参考資料

LPC185x-Xplorer++

About NGX Technologies

NGX Technologies is a premier supplier of development tools for the ARM7, ARM Cortex M0, M3 and M4 series of microcontrollers. NGX provides innovative and cost effective design solutions for embedded systems. We specialize in ARM MCU portfolio, which includes ARM7, Cortex-M0, M3 & M4 microcontrollers. Our experience with developing evaluation platforms for NXP controller enables us to provide solutions with shortened development time thereby ensuring reduced time to market and lower development costs for our customers. Our cost effective and feature rich development tool offering, serves as a testimony for our expertise, cost effectiveness and quality.

Contact Information:

NGX Technologies Pvt. Ltd.
No.216, 5th main Road, R.P.C. Layout,
Vijayanagar 2nd Stage,
Bangalore – 560 104
Phone : +91-80-40925507
email:sales@ngxtechnologies.com

CE certification:

NGX Technologies LPC185X-Xplorer++ board has been tested for radiated emission as per EN55022 class a standard. The device is under the limits of the standard EN55022 class A and hence CE marked. No other test have been conducted other than the radiated emission (EN55022 class A standard). The device was tested with the ports like USB, Serial, and Power excluding the GPIO ports. Any external connection made to the GPIO ports may alter the EMC behavior. Usage of this device under domestic environment may cause unwanted interference with other electronic equipment's. User is expected to take adequate measures. The device is not intended to be used in and end product or any subsystem unless the user re-evaluates applicable directive/conformance.

Table of Contents

1.0 INTRODUCTION....4
2.0 LPC185X-Xplorer++ Development Tool Setup....5
2.1 ULINK2/ME and KEIL....5
2.2 Installation & Configuration of KEIL software....5
2.3 Configuration of ULINK2/ME Debugger....9
2.4 Configuring External Quad Flash....10
2.5 Setup for ULINK2/ME and LPC185X-Xplorer++ Board....12
3.0 LPC185X-Xplorer++ firmware Development....17
3.1 Creating the sample (Blinky) project in KEIL....17
3.2 Executing the sample project in KEIL....34
4.0 Creating bin file....36
5.0 LPC185x-Xplorer++ Programming....37
5.1 Programming options....37
5.1.1 Programming LPC185x-Xplorer++ using Coflash and ColinkEx....37
6.0 Restoring LPC185X-Xplorer++ to Factory Defaults....40
6.1 ULINK2/ME and KEIL....40
7.0 Schematic & Board Layout....41
7.1 Schematic....41
7.2 Board layout....41
8.0 CHANGE HISTORY....42
8.1 Change History....42

1.0 INTRODUCTION

This document is the User Manual for LPC185X-Xplorer++; a cost effective evaluation platform for NXP's LPC185x MCU. This document reflects its contents which include system setup, debugging, and software components. This document provides detailed information on the overall design and usage of the board from a systems perspective.

Before proceeding further please refer the Quick Start Guide for LPC185X-Xplorer++ features, Unboxing and verification. Kindly refer to the product page for the latest information.

Note: To restore the Factory Default for 'LPC185X-Xplorer++ Board' kindly refer to section 6.0

2.0 LPC185X-Xplorer++ Development Tool Setup

2.1 ULINK2/ME and KEIL

NGX's MCU evaluation platforms are not coupled tightly with any one particular combination of IDE and debugger. The following sections will explain the setup for KEIL and ULINK2/ME as the IDE and debugger respectively.

2.2 Installation & Configuration of KEIL software

The Installation of KEIL (MDK-ARM V4.54) software is explained below:

Step 1: Open the KEIL setup

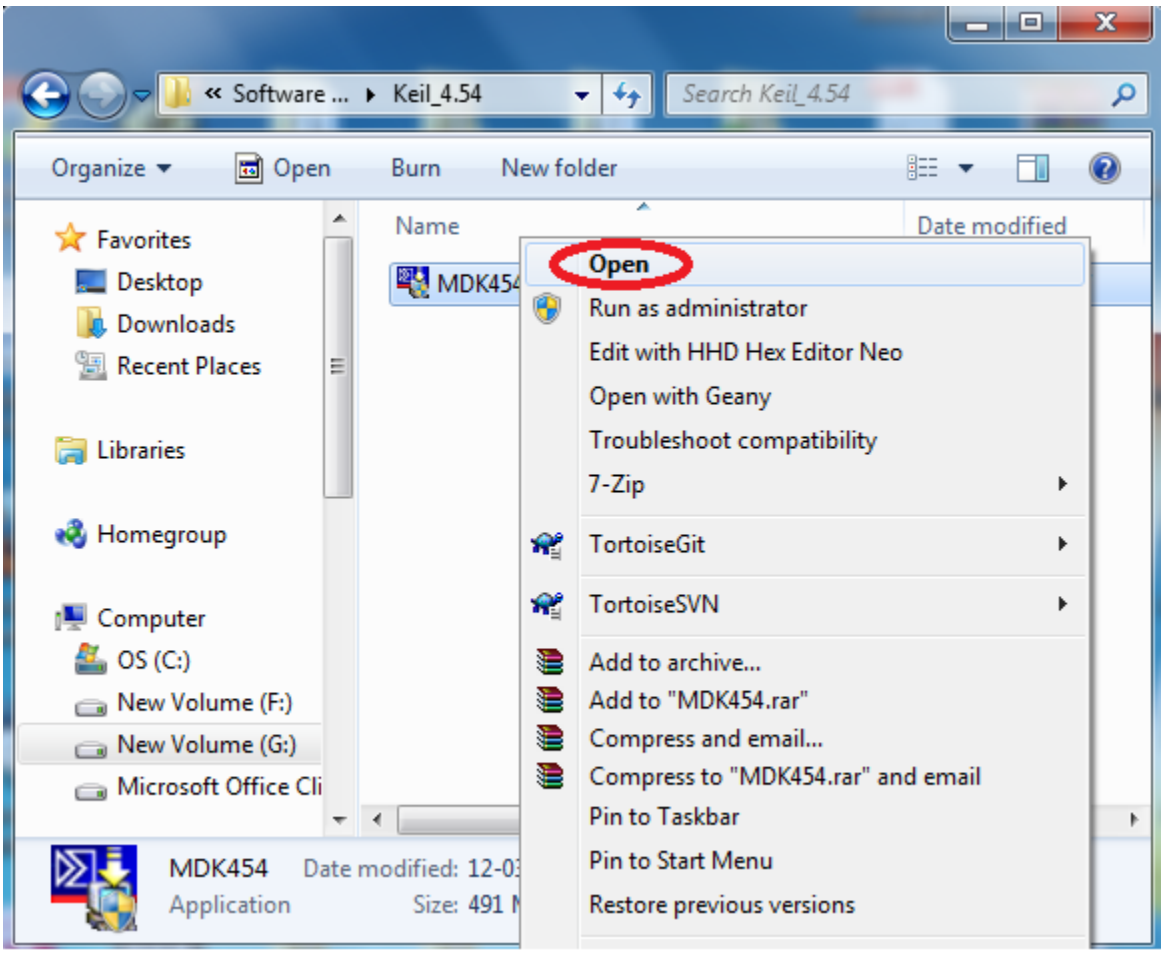


Fig.1

Step 2: Click on Next.

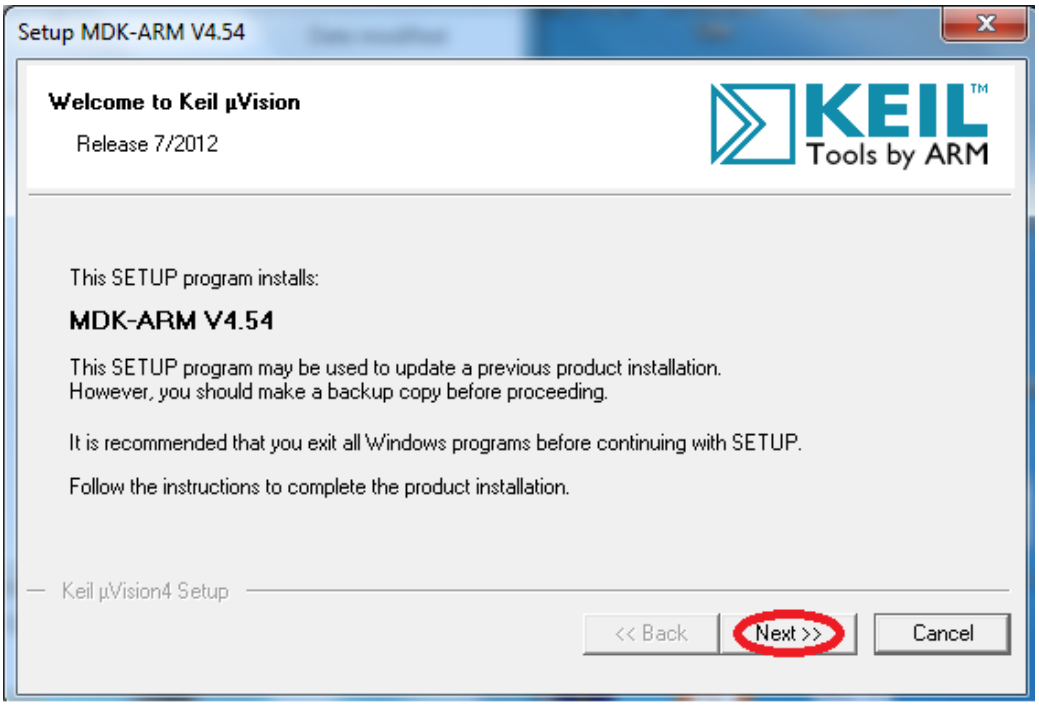


Fig.2

Step 3: To accept the agreement, click the check box and click Next.

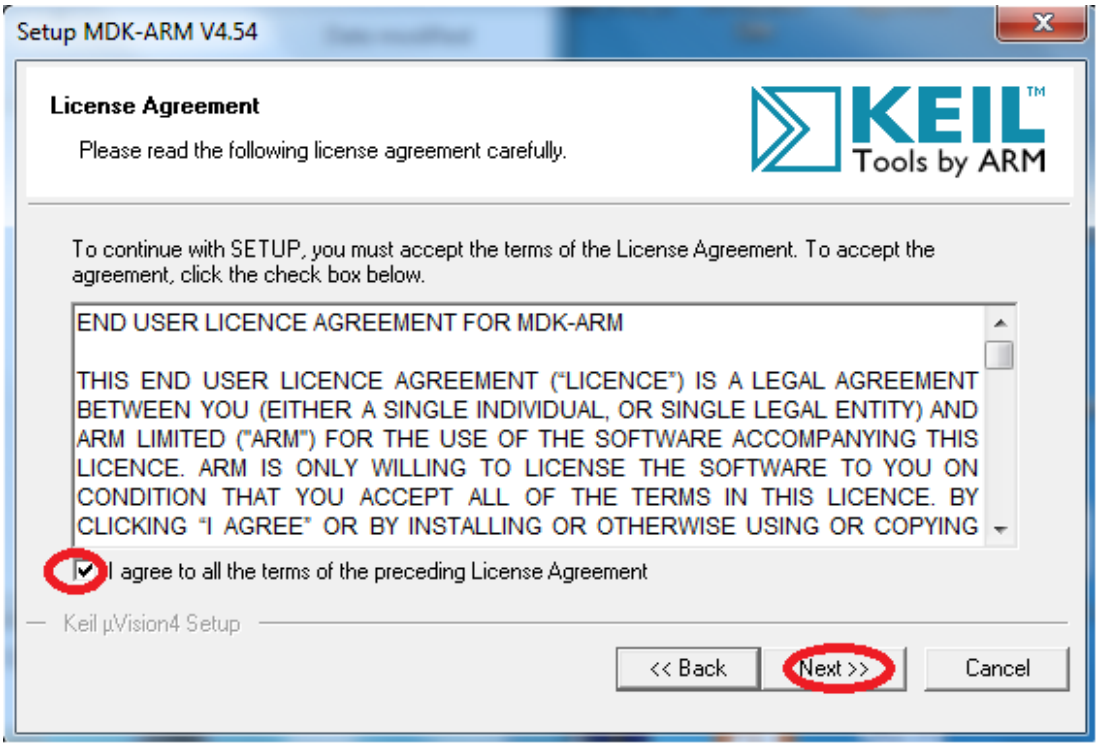


Fig.3

Step 4: Provide the destination path and click on Next

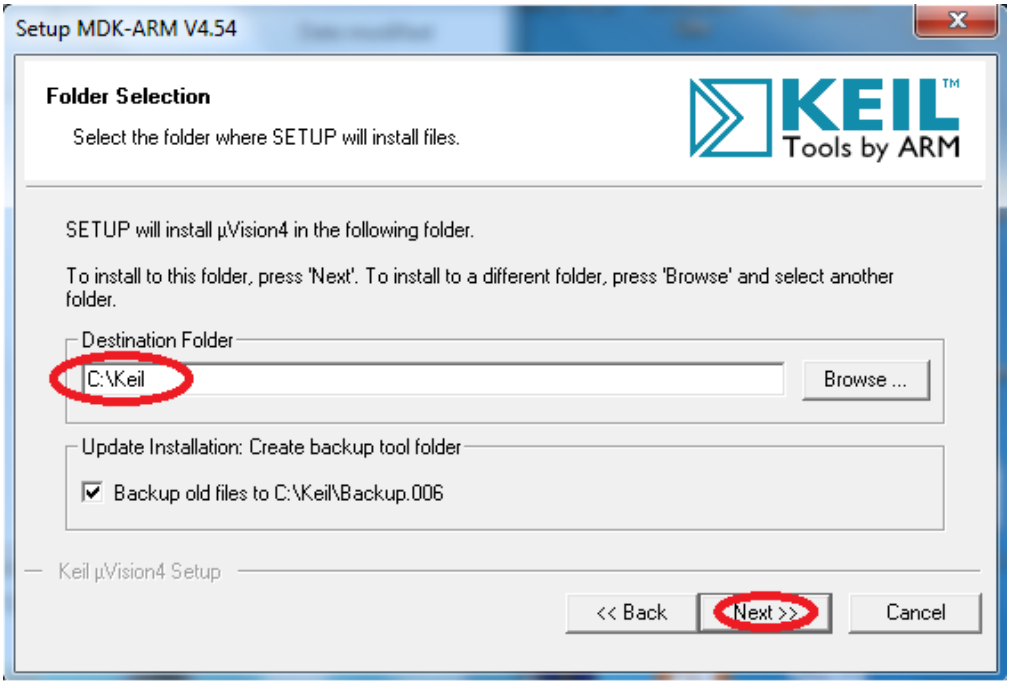


Fig.4

Step 5: Fill your personal information and click on Next

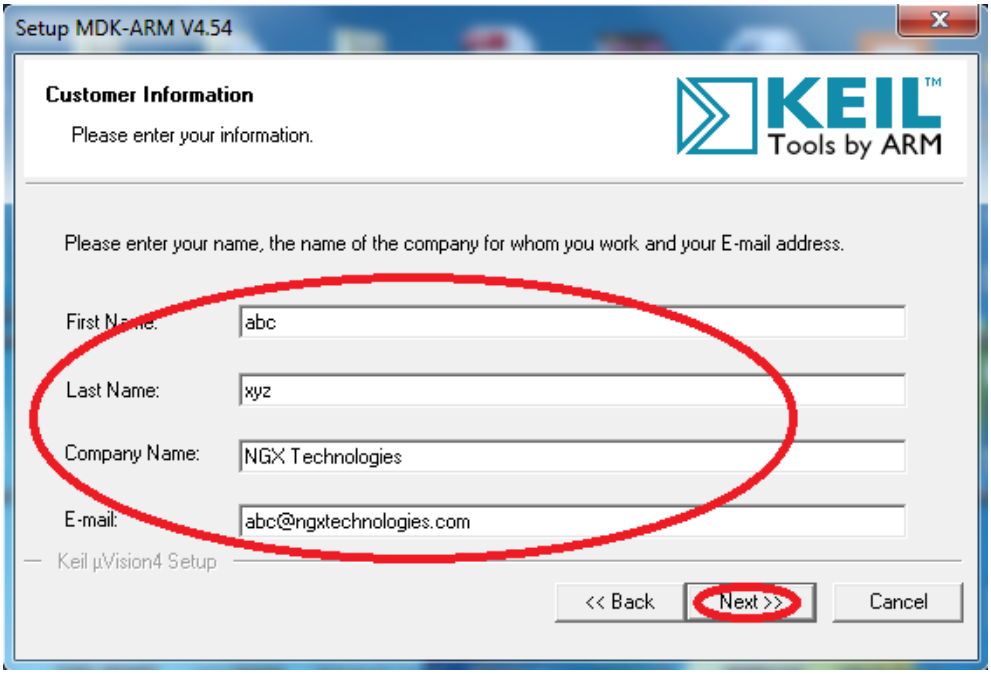


Fig.5

Step 6: Click on Next

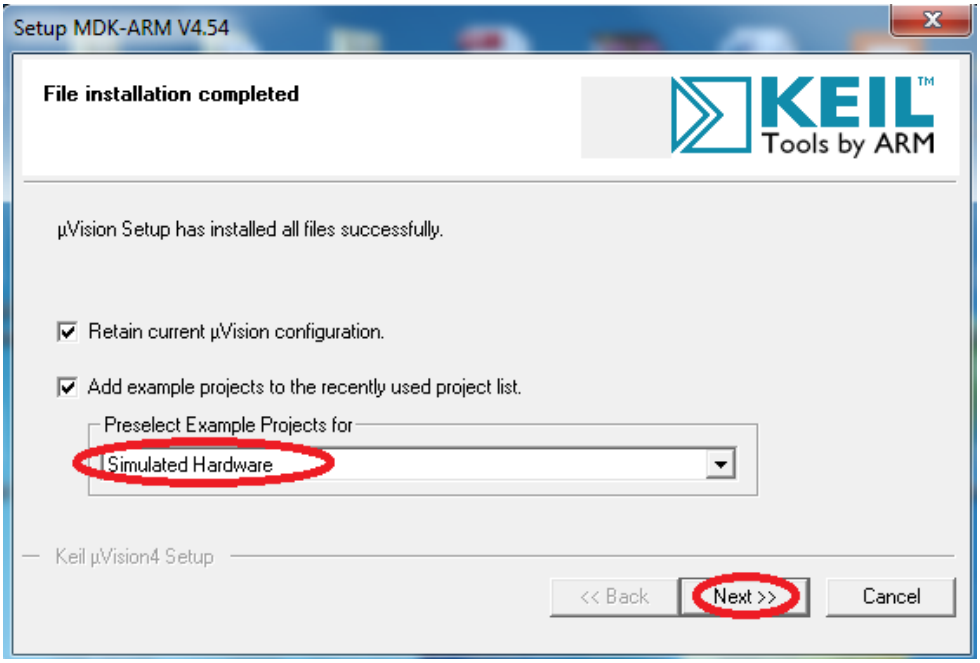


Fig.6

Step 7: Keil µVision4.54 setup is completed, click Finish.

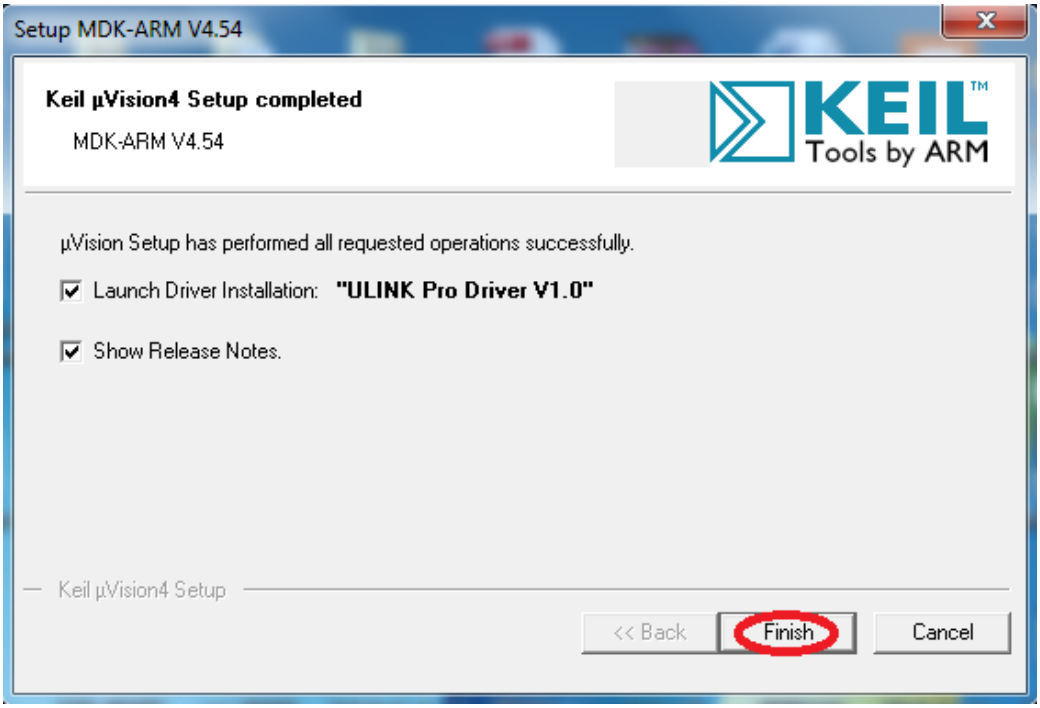


Fig.7

2.3 Configuration of ULINK2/ME Debugger

The configuration flow of ULINK2/ME Debugger is explained below:

Step 1: Open the KEIL blinky project downloaded from the website and click on the 'Target Options'.

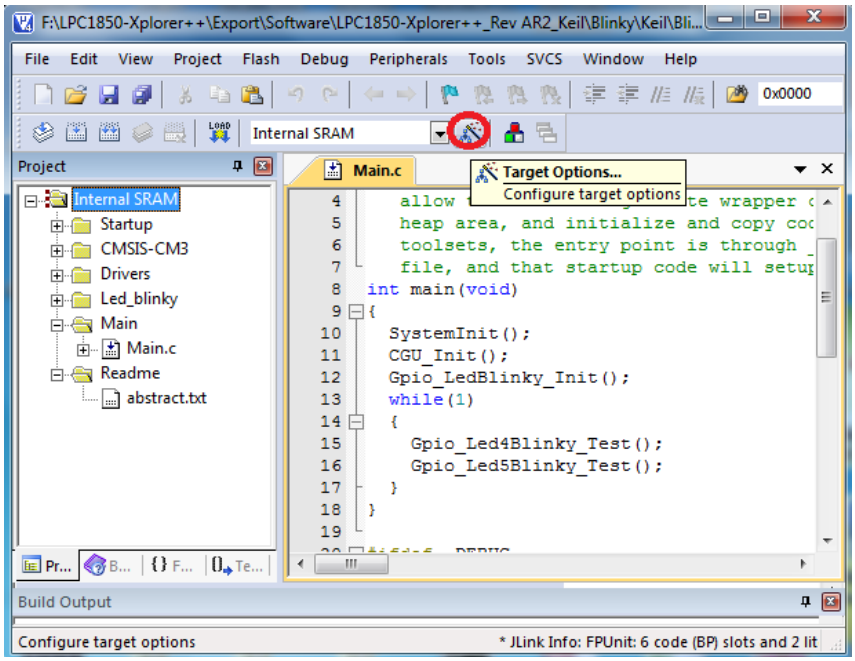


Fig.8

Step 2: The window opens as shown in the following image. Next click on Debug and then select the 'ULINK2/ME Cortex Debugger' as shown in the following image.

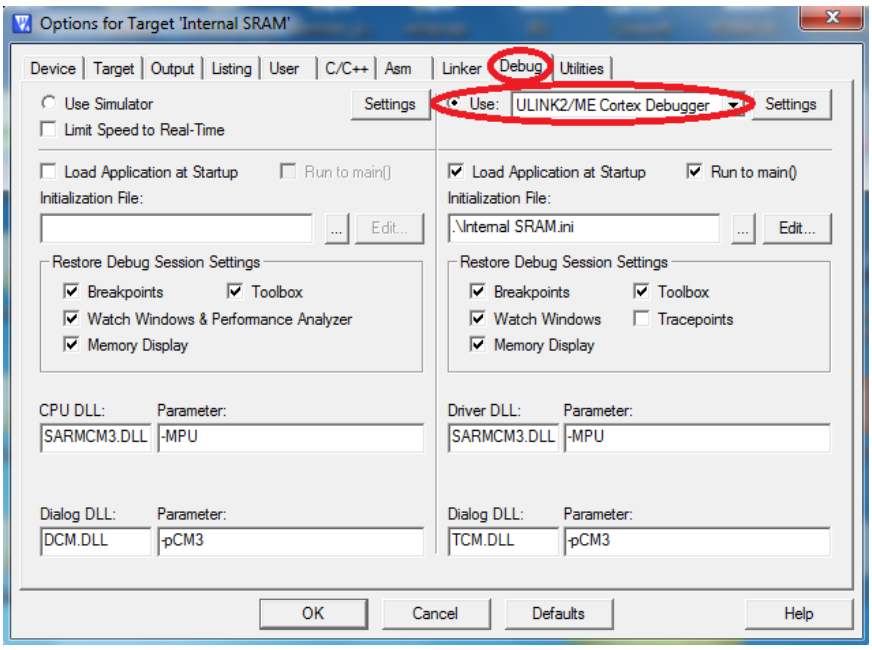


Fig.9

Step 3: Next click on the 'Settings Option', the 'Cortex-M Target Driver Setup' window opens then select SW port. After selection of the SW port the ULINK2/ME detected is as shown in the following image.

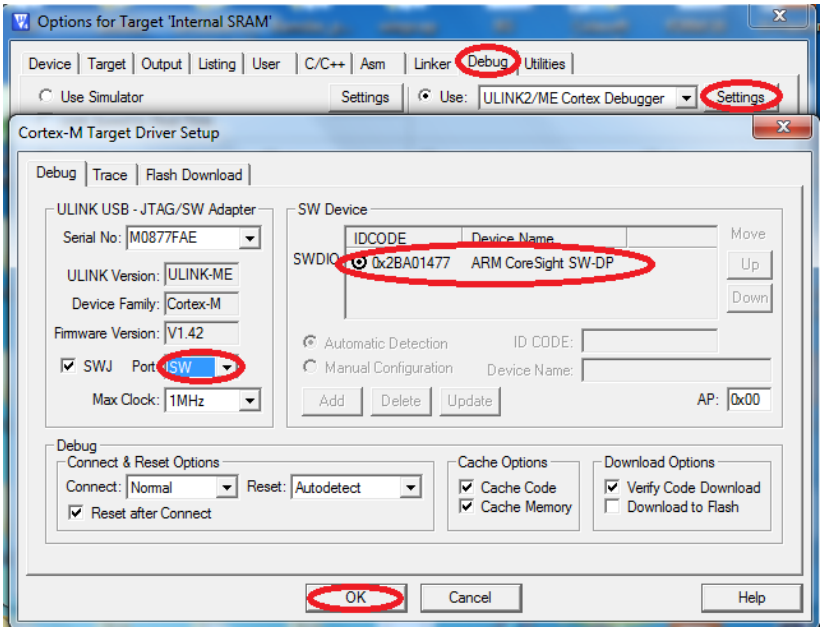


Fig.10

(Note: The Cortex M3 can be programming using SW or JTAG.)

2.4 Configuring External Quad Flash

Step 1: Open the KEIL blinky project, and then click on the 'Target Options'.

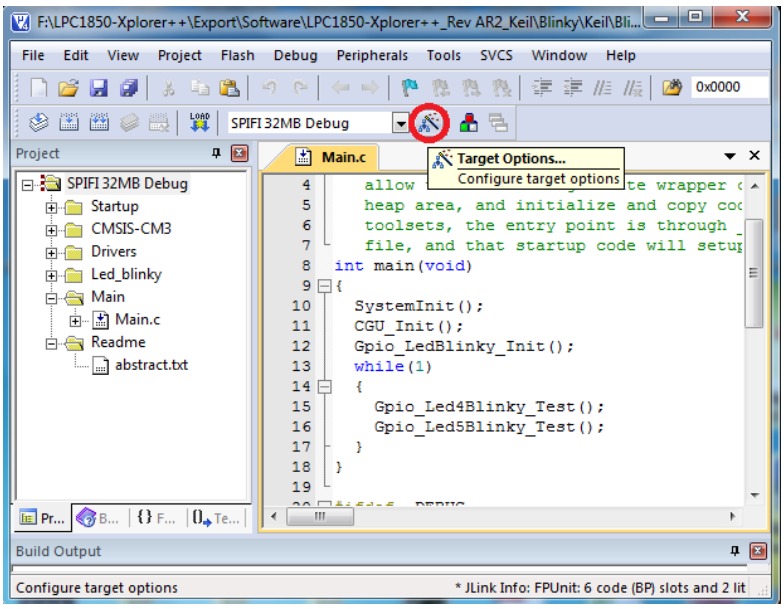


Fig.11

Step 2: The window opens as shown in the following image, click on Utilities and select ‘ULINK2/ME Cortex Debugger’.

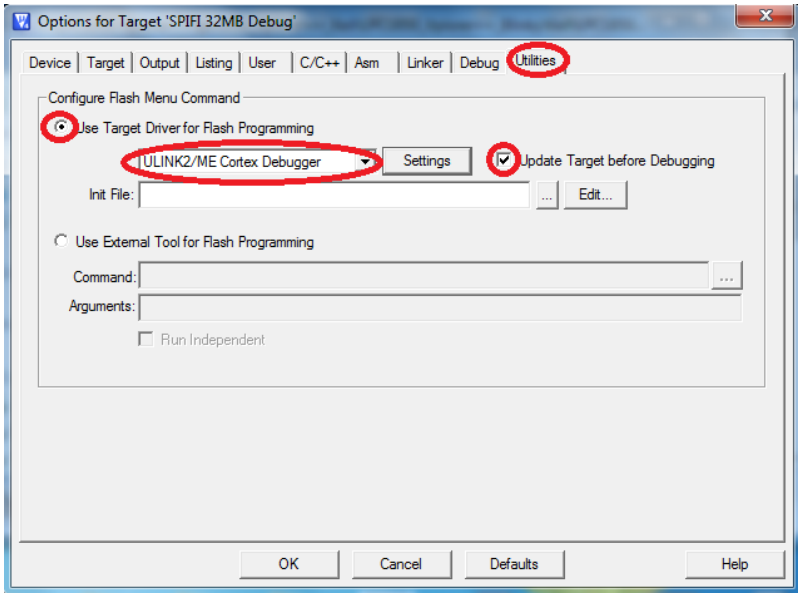


Fig.12

Step 2: Select ‘Update Target before Debugging’ check box, click on ‘Settings Options’ and in ‘Cortex-M Target Driver setup’ click add and select ‘LPC18xx/43xx S25FL032 SPIFI programming algorithm’ and click add.

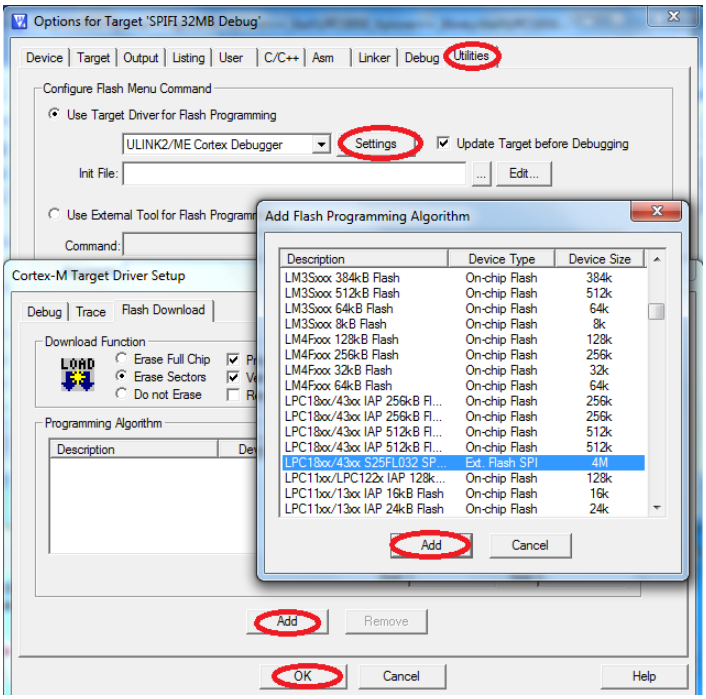


Fig.13

Click OK to complete the ‘ULINK2/ME Debugger configuration’.

2.5 Setup for ULINK2/ME and LPC185X-Xplorer++ Board

Option A: With ULINK-ME debugger

To run the KEIL examples you will need the following and the image shows the each components:

- ULINK-ME
- 10-pin ribbon cable
- LPC185X-Xplorer++ Board
- Two USB AM to Micro B cable

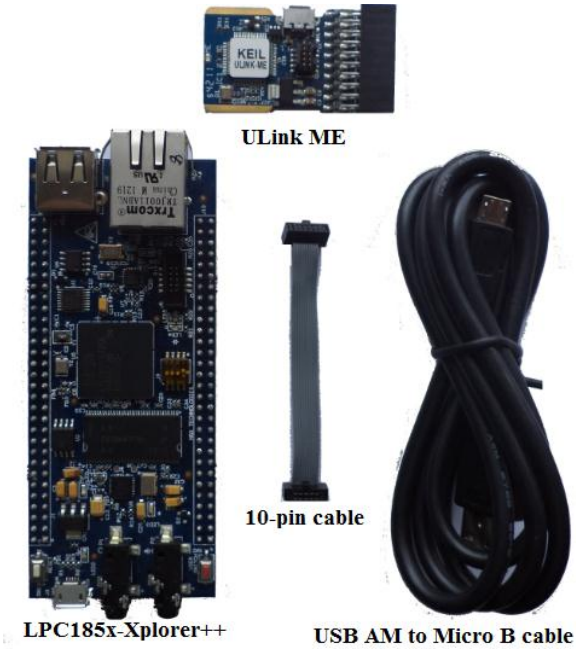


Fig.14

Steps to setup the ULINK-ME and LPC185X-Xplorer++ Board:
(Note: Please refer keil knowledgebase article for connecting ULINK2/ME 10-pin ribbon cable to NGX Xplorer++)

Step 1: Connect one end of 10-pin ribbon cable to 'ULINK-ME 10-pin box header' as show in the following image.

Fig.15

Step 2: Connect other end of 10-pin ribbon cable to ‘10-pin box header’ of the LPC185X-Xplorer++ board as shown in the following image.

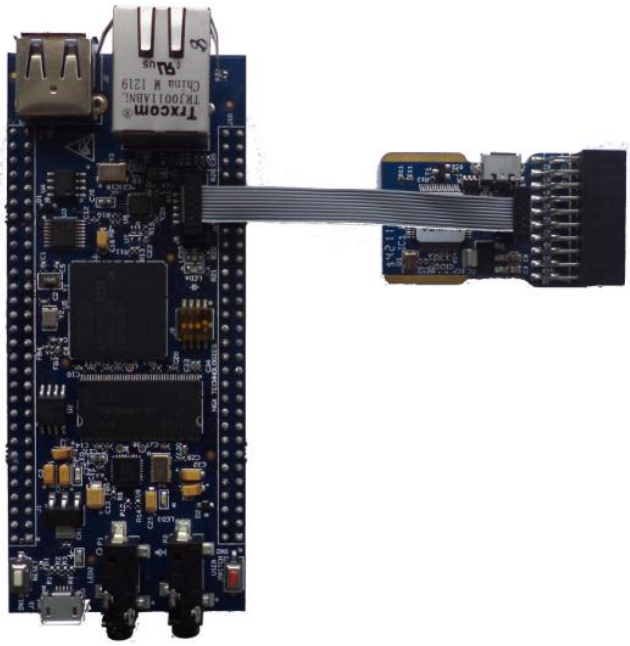


Fig.16

Step 4: Connect one end of ‘USB AM to Micro B’ cable to LPC185X-Xplorer++ board and other end to computer and connect one end of ‘USB AM to Micro B’ to ULINK-ME and other end to computer as shown in the following image.

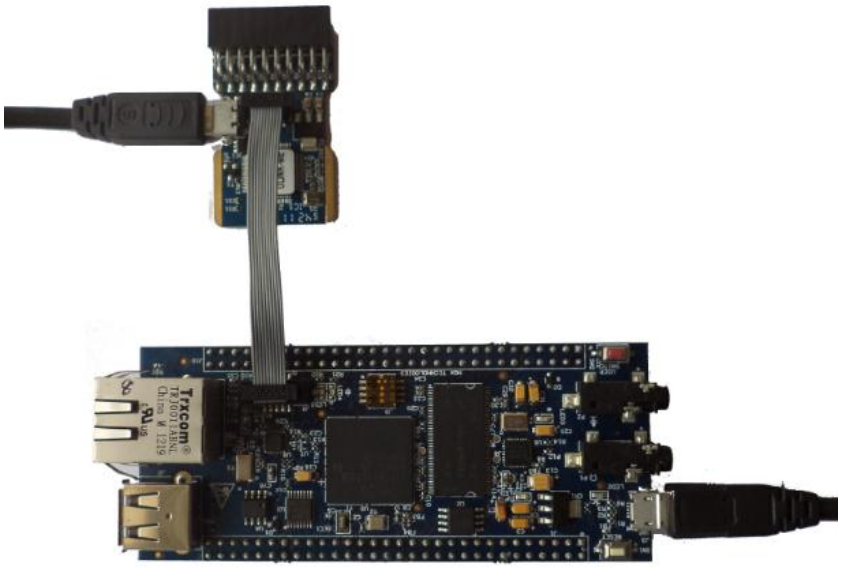


Fig.17

The setup is now ready to be used for development with **KEIL IDE** and **ULINK-ME**.

Option B: With ULINK2 debugger

The LPC185X-Xplorer++ board has on board '10-pin SWD/JTAG box header'; ensure that the ULINK2/ME must have '10-pin SWD/JTAG' support for development. The ULINK2 debugger, '20-pin to 10-pin adapter' and 10-pin ribbon cable are not a part of the LPC185X-Xplorer++ package, the user needs to buy separately.

To run the KEIL examples you will need the following and the image shows the each components:

- ULINK2
- ARM JTAG to Cortex JTAG Adapter (20-pin to 10-pin Adaptor)
- 10-pin ribbon cable
- LPC185X-Xplorer++ Board
- One USB AM to Micro B cable
- One USB type B cable

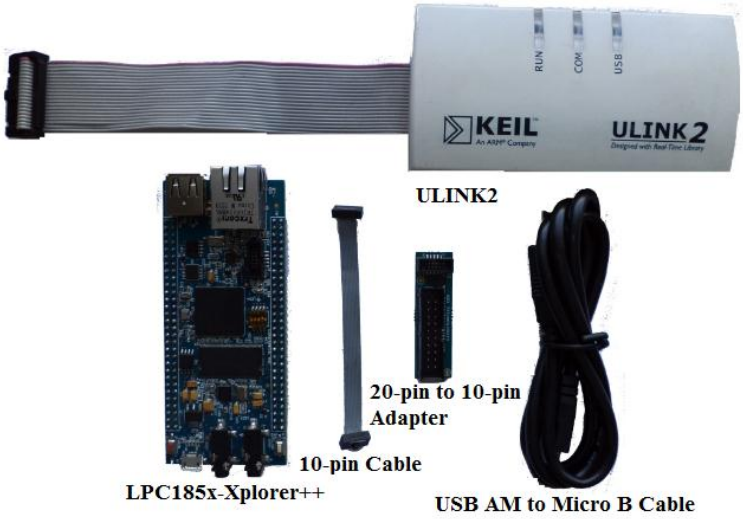


Fig.18

Steps to setup the ULINK2 and LPC185X-Xplorer++ Board:
(Note: Please refer keil knowledgebase article for connecting ULINK2/ME 10-pin ribbon cable to NGX Xplorer++)

Step 1: Connect the one end of 10-pin ribbon cable to '20-pin to 10-pin adapter' as shown in the following image.

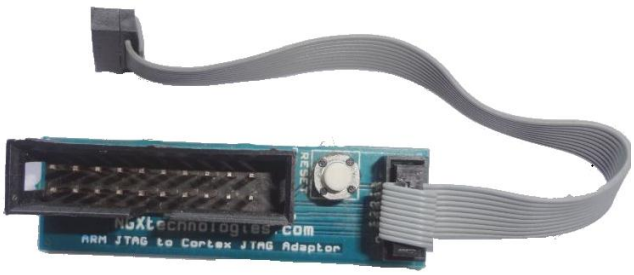


Fig.19

Step 2: Connect other end of 10-pin ribbon cable to ‘10-pin box header’ of the LPC185X-Xplorer++ board as shown in the following image.

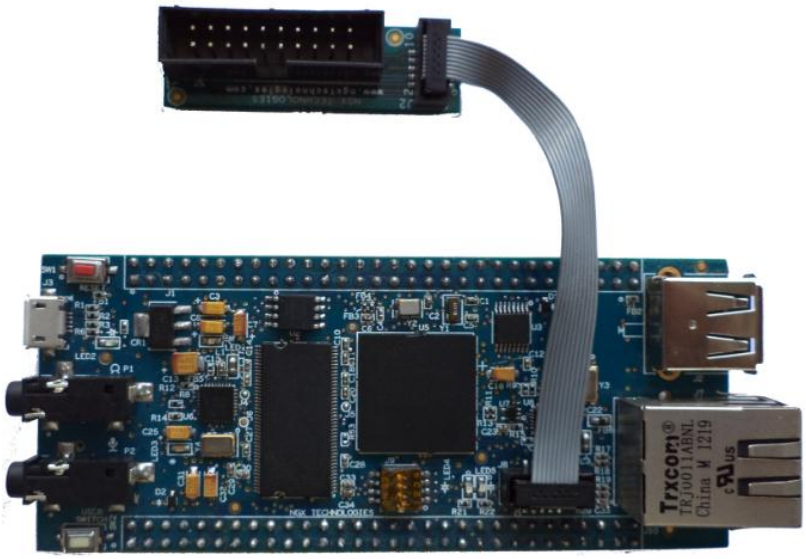


Fig.20

Step 3: Now connect the ‘ULINK2 20-pin’ cable to ‘20-pin to 10-pin adapter’ as shown in the following image.

Fig.21

Step 4: Connect one end of ‘USB AM to Micro B’ cable to LPC185X-Xplorer++ board and other end to computer, connect one end of ‘USB type B’ to ULINK2 and other end to computer as shown in the following image.

Fig.22

The setup is now ready to be used for development with **KEIL IDE** and **ULINK2**.

3.0 LPC185X-Xplorer++ firmware Development

3.1 Creating the sample (Blinky) project in KEIL

Steps to create the sample (Blinky) project:

Step 1: Open a KEIL IDE.

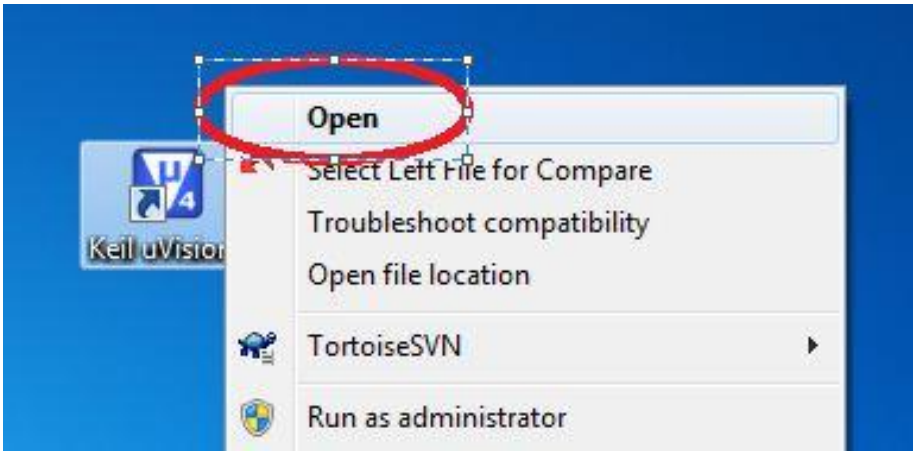


Fig.23

Step 2: Click on Project->New uVision Project… as show below.

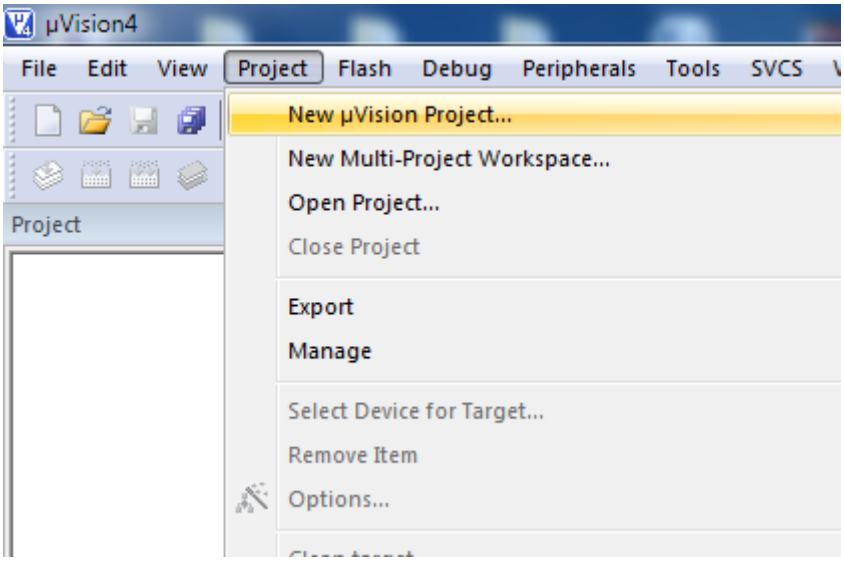


Fig.24

Step 3: Create a new folder in downloaded sample example folder and rename to Blinky_test and select Blinky_test folder click Open.

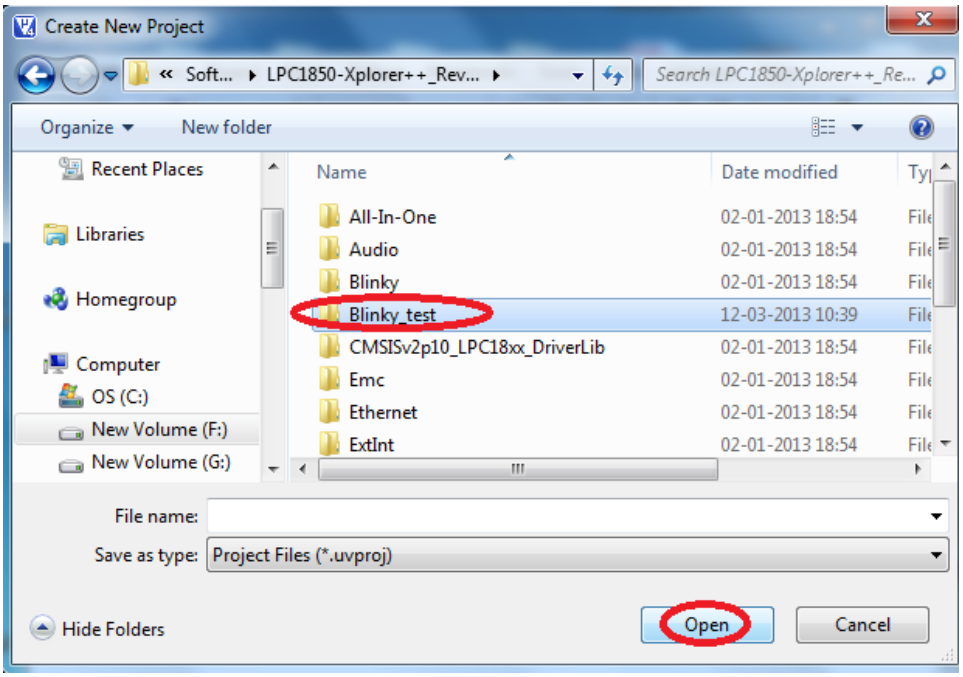


Fig.25

Step 4: Give a project name, example: 'Blinky_test' and click Save.

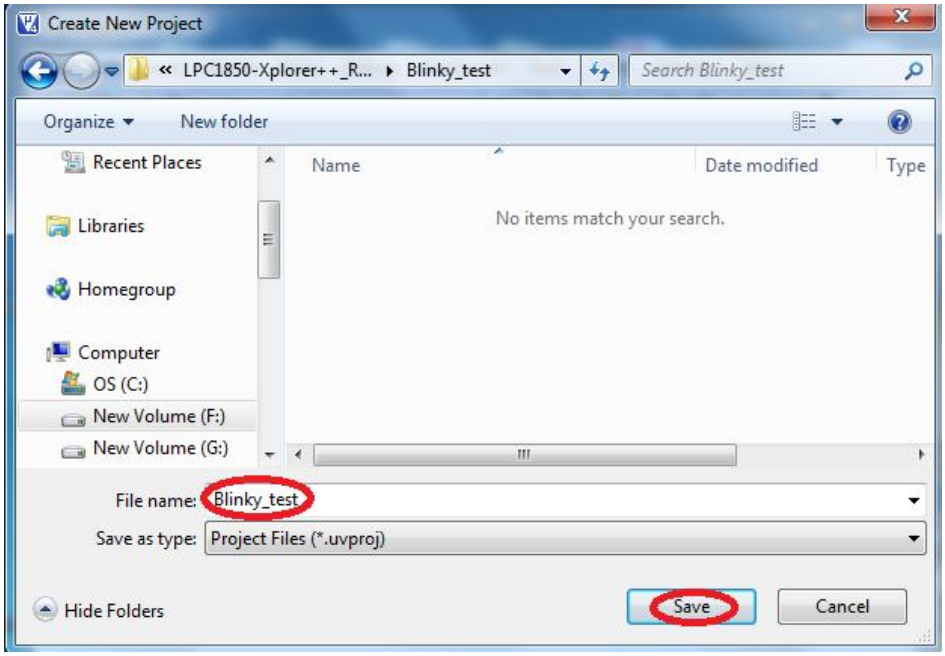


Fig.26

Step 5: Select the NXP (found by Philips) and search for the controller.

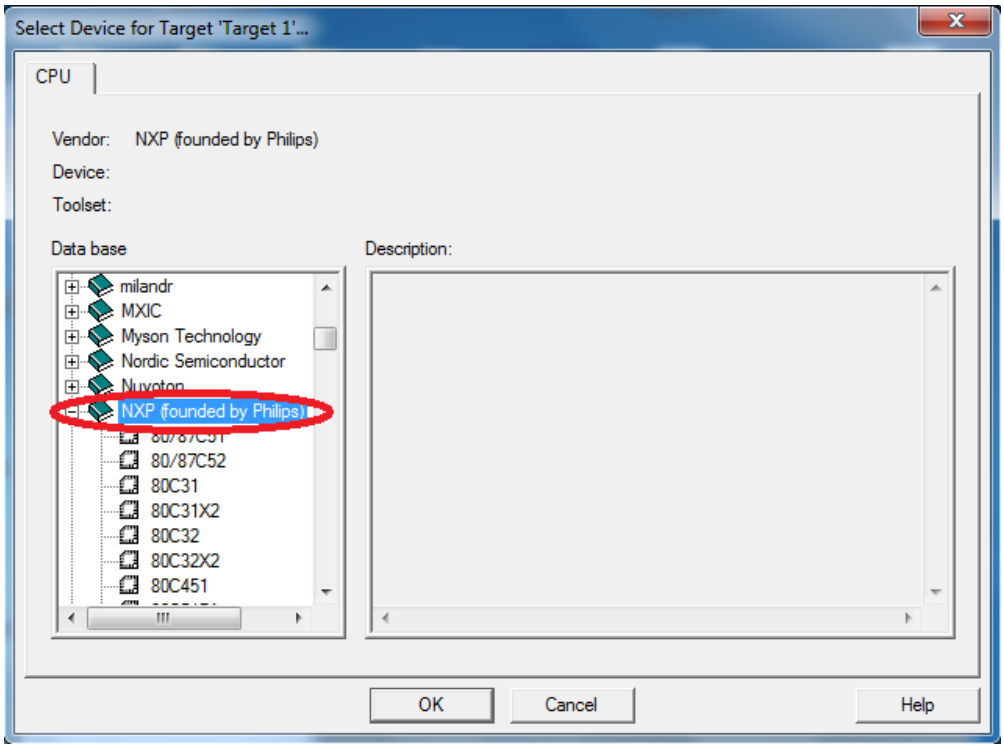


Fig.27

Step 6: Select LPC185X controller (Select LPC1850 for LPC1850-Xplorer++ board, LPC1857 for LPC1857-Xplorer++ board) and click OK.

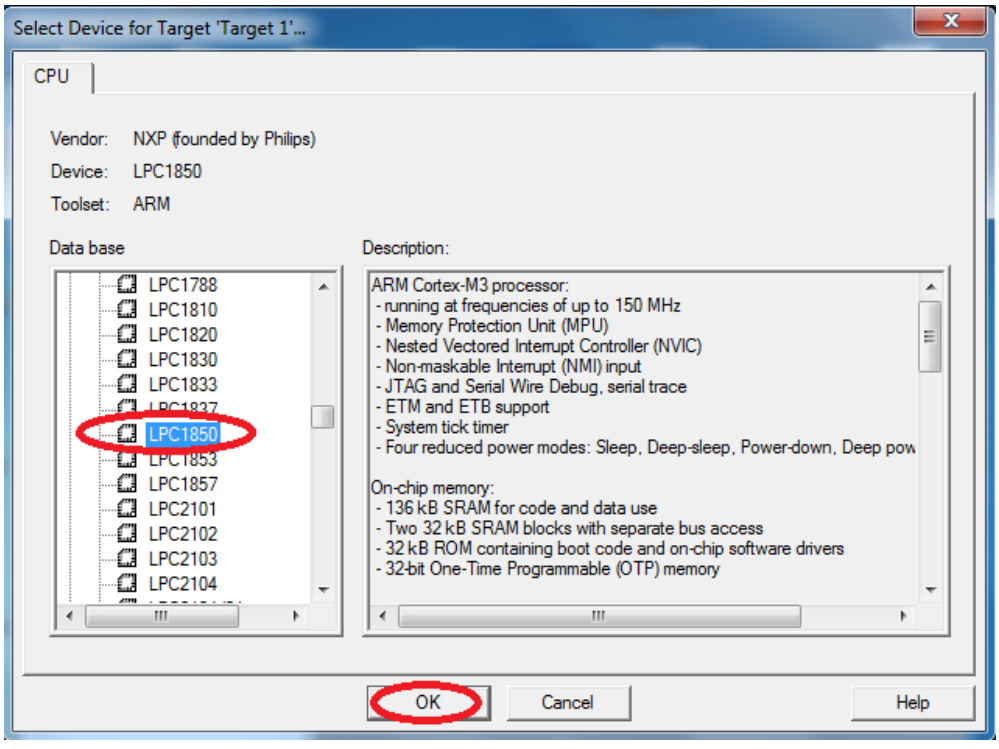


Fig.28

Step 7: Click YES to copy startup file to project folder and add file to project.

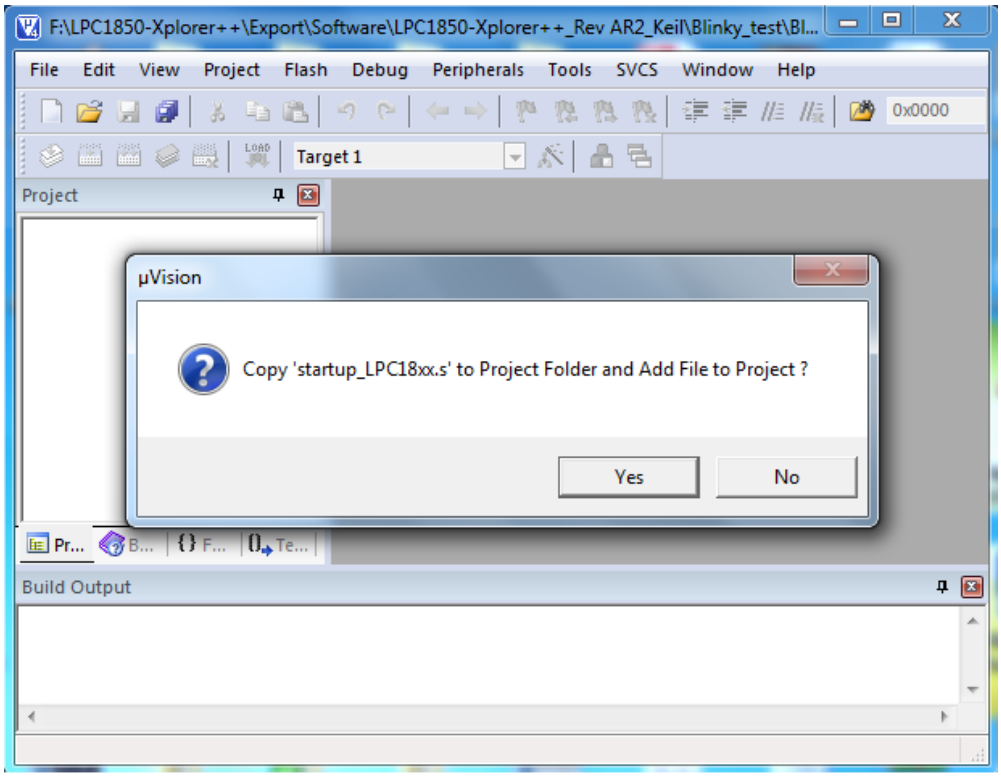


Fig.29

Step 8: Rename the “Target1” to “Blinky_test” and “Source Group1” to “Startup”.

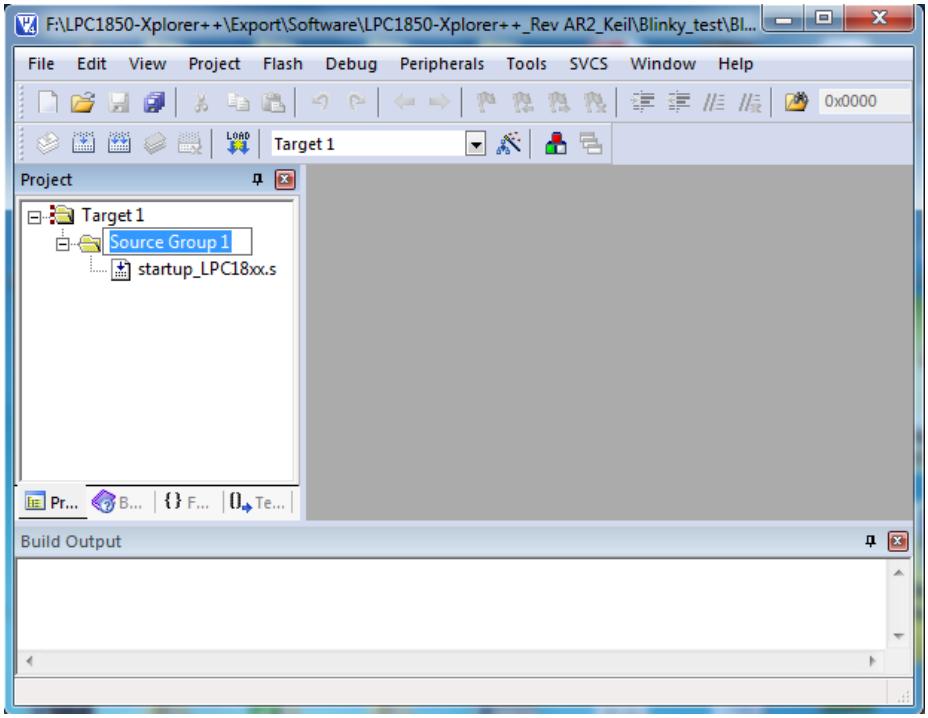


Fig.30

Step 9: Double click on Startup.

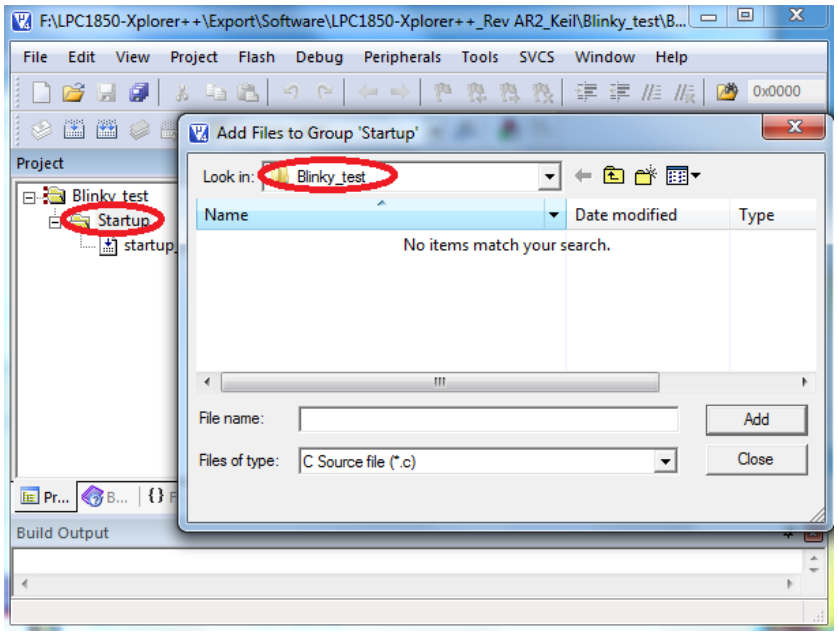


Fig.31

Step 10: Search 'system_LPC18xx.c file' (in the sample examples downloaded folder from NGX website), the file will be found at following path: **"..\CMSISv2p10_LPC18xx_DriverLib\Core\Device\NXP\LPC18xx\Source\Templates"** select 'system_LPC18xx.c file' and click Add as shown in the following image.

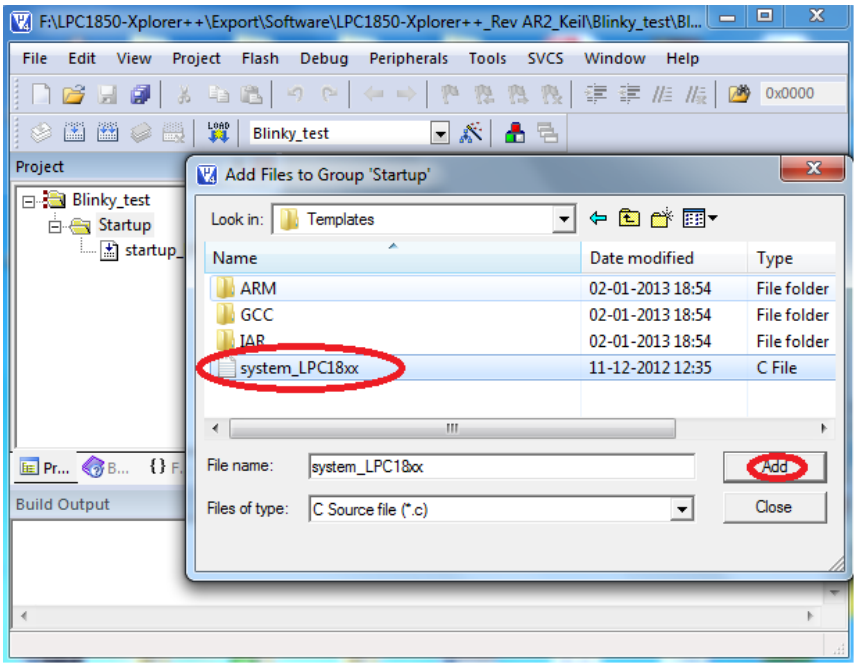


Fig.32

Delete Startup_LPC18xx.s file and add the Startup_LPC18xx.s from the following path **\CMSISv2p10_LPC18xx_DriverLib\Core\Device\NXP\LPC18xx\Source\Templates\ARM**

Step 10: Right click on Blinky_test to add a new group.

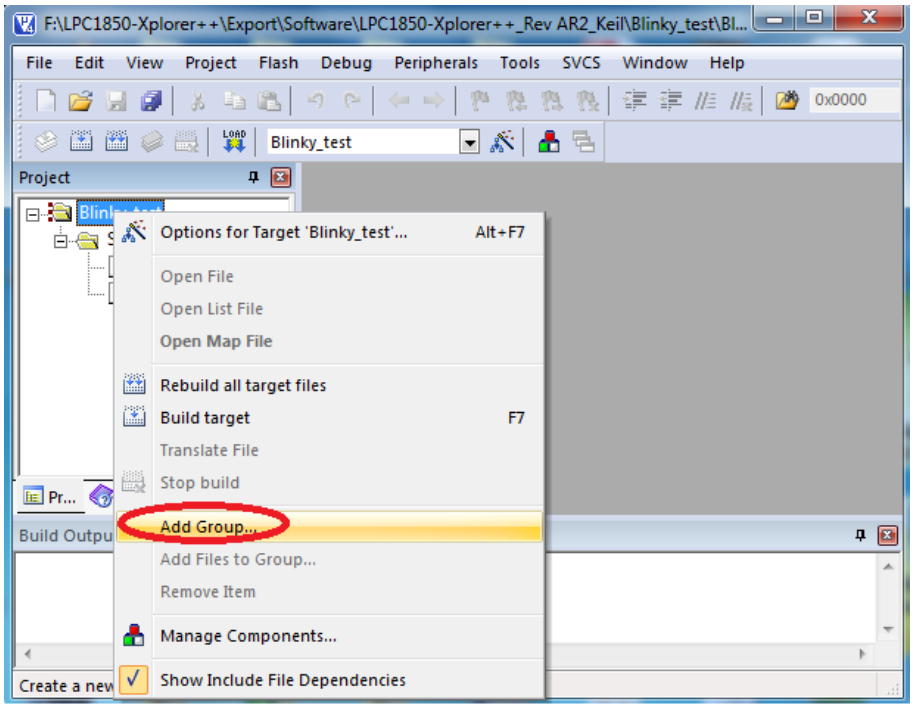


Fig.33

Step 11: Rename the 'New Group' to Drivers.

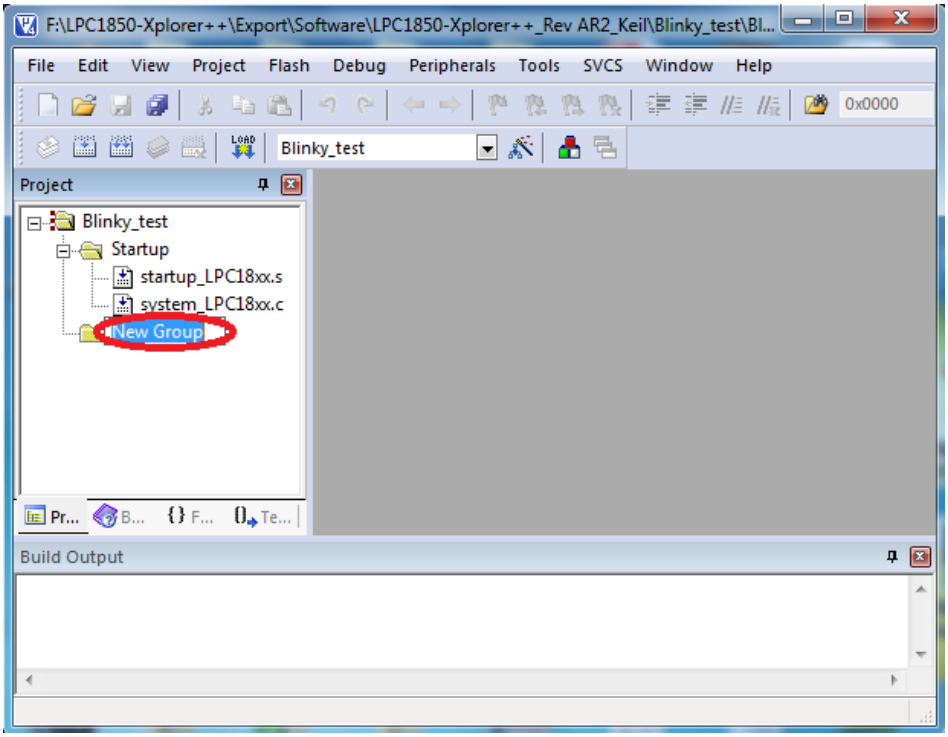


Fig.34

Step 12: Double click on Drivers and search the driver files (in the sample examples downloaded folder from NGX website) the driver files will be found at following path: **'..\CMSISv2p10_LPC18xx_DriverLib\src'** for blinky project we have to select lpc18xx_cgu.c, lpc18xx_gpio.c, lpc18xx_scu.c, lpc18xx_timer.c and lpc18xx_utils.c driver files and click Add.

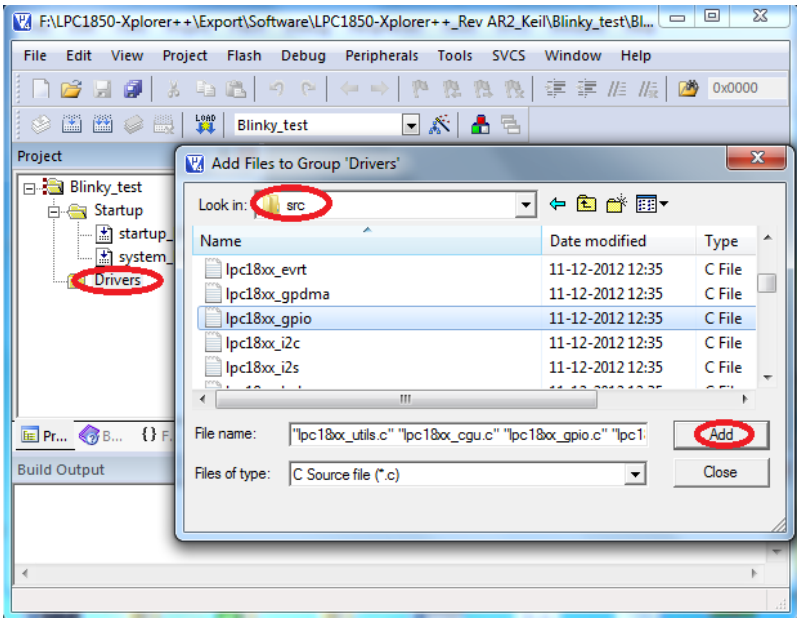


Fig.35

Step 13: Create another new group and rename it as Main.

Step 14: Click on New to create an empty document.

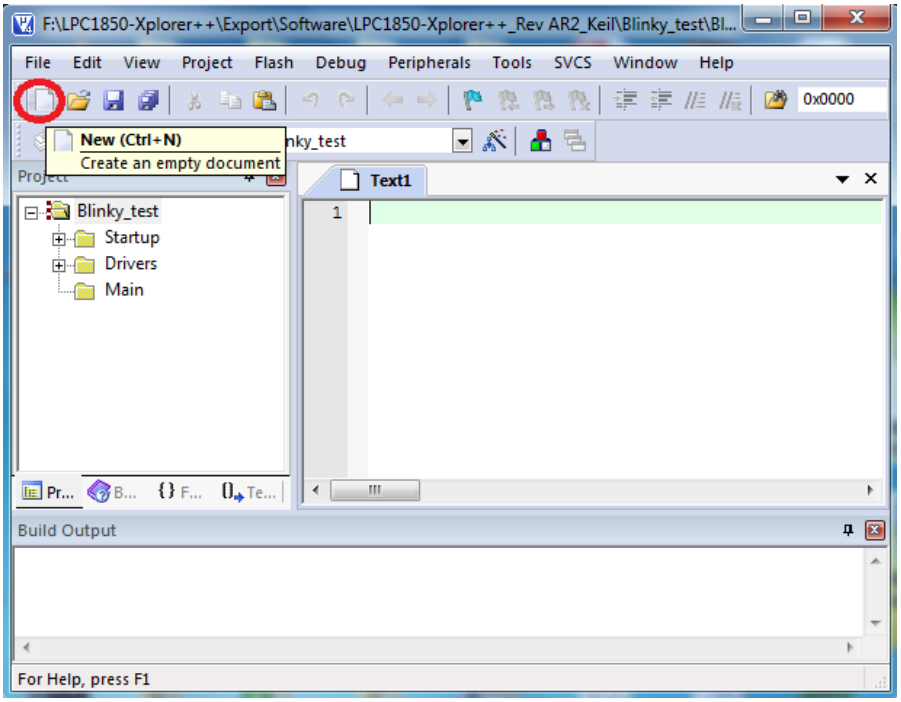


Fig.36

Step 15: Implement the C instructions need to blink a LED on LPC185x-Xplorer++ and save it to main.c in 'Blinky_test folder' as shown in the following image.
(Note: Please refer Downloaded 'Blinky' example)

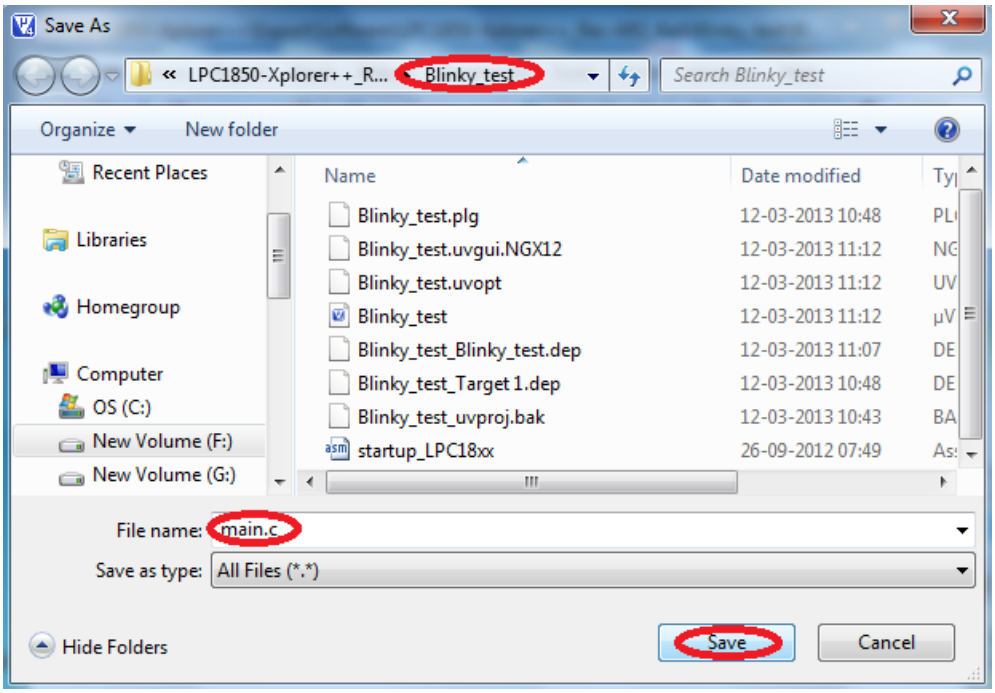


Fig.37

Step 16: Double click on Main Group, and select main.c and click on Add

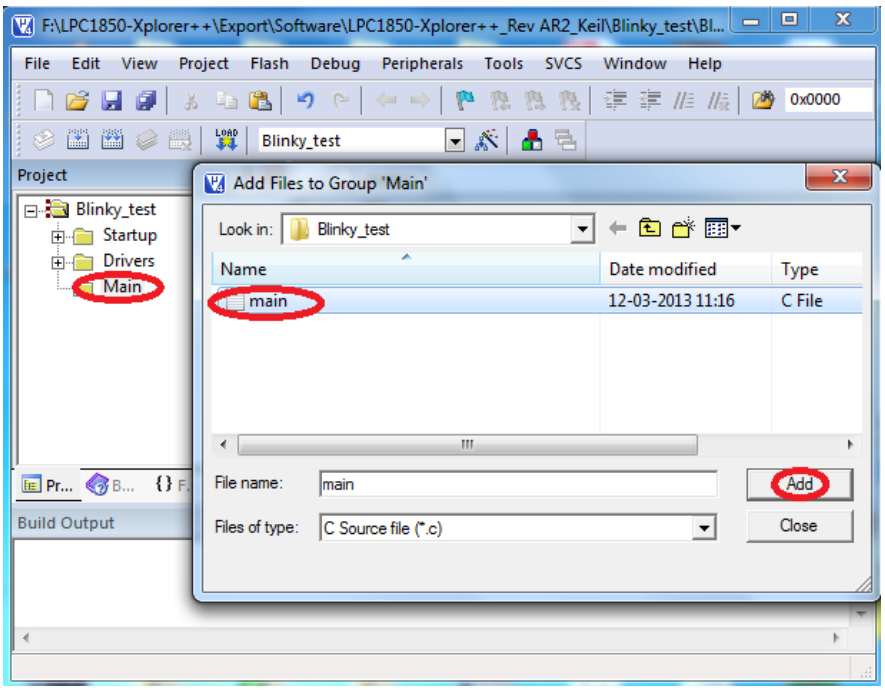


Fig.38

Step 17: Click ‘Target Options’, make following changes for ‘Internal SRAM’ as shown in the following image.

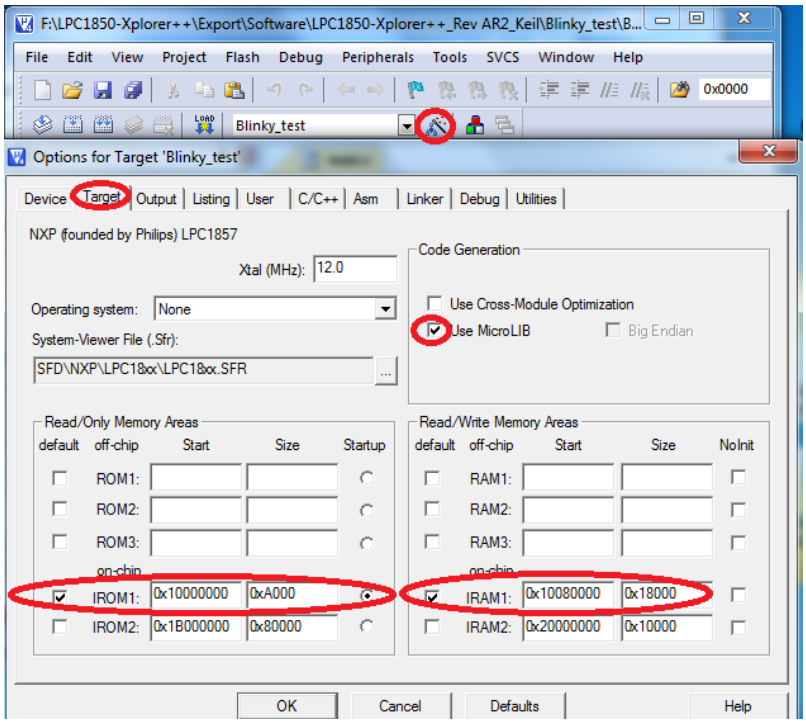


Fig.39

For ‘External SPIFI Flash’, make following changes as shown in the following image.

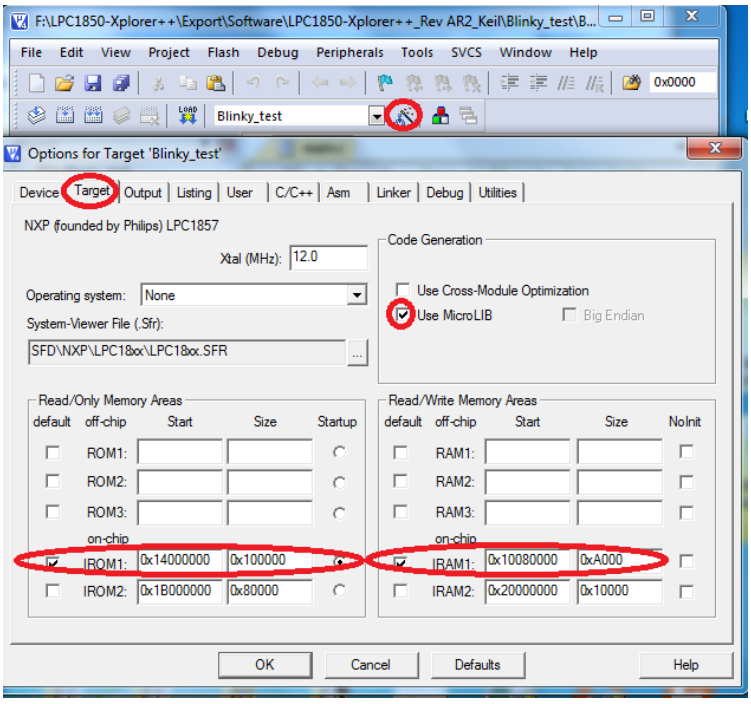


Fig.40

Step 18: In ‘Output Options’ select ‘Create HEX File’ check box.

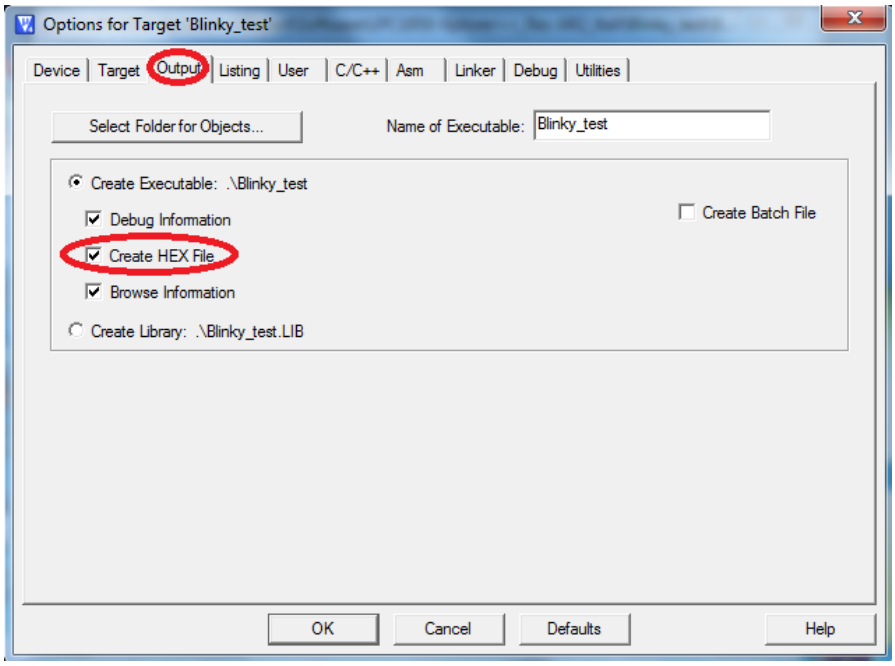


Fig.41

Step 19: Click ‘C/C++ option’, in Define type ‘CORE_M3’, click on ‘Include paths’ to include Drivers header file path as shown in the following image.

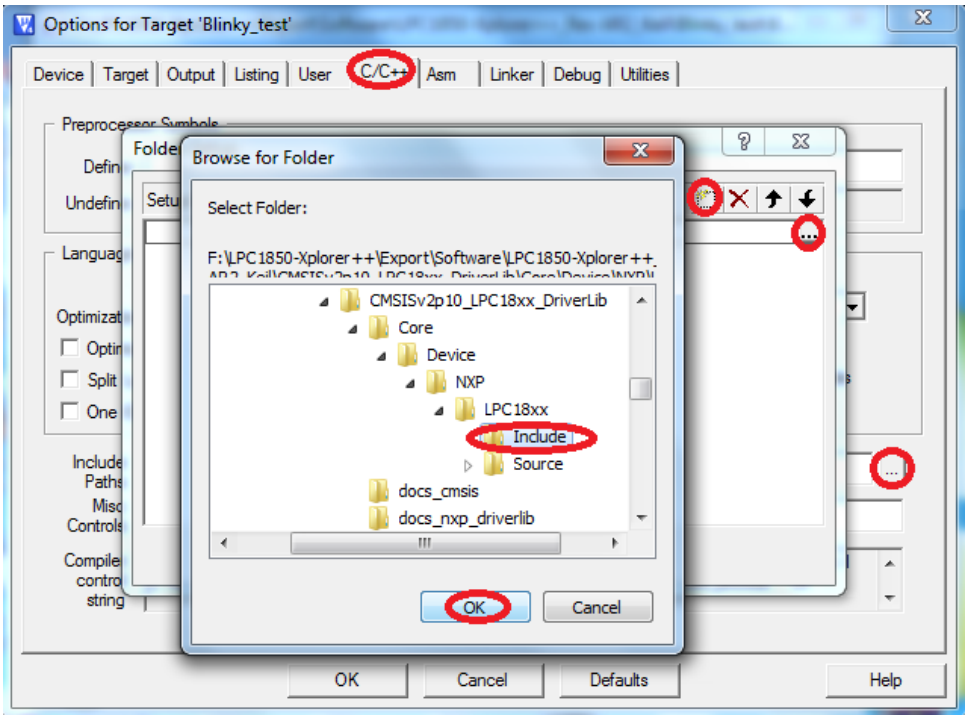


Fig.42

Include all paths as shown in bellow image.

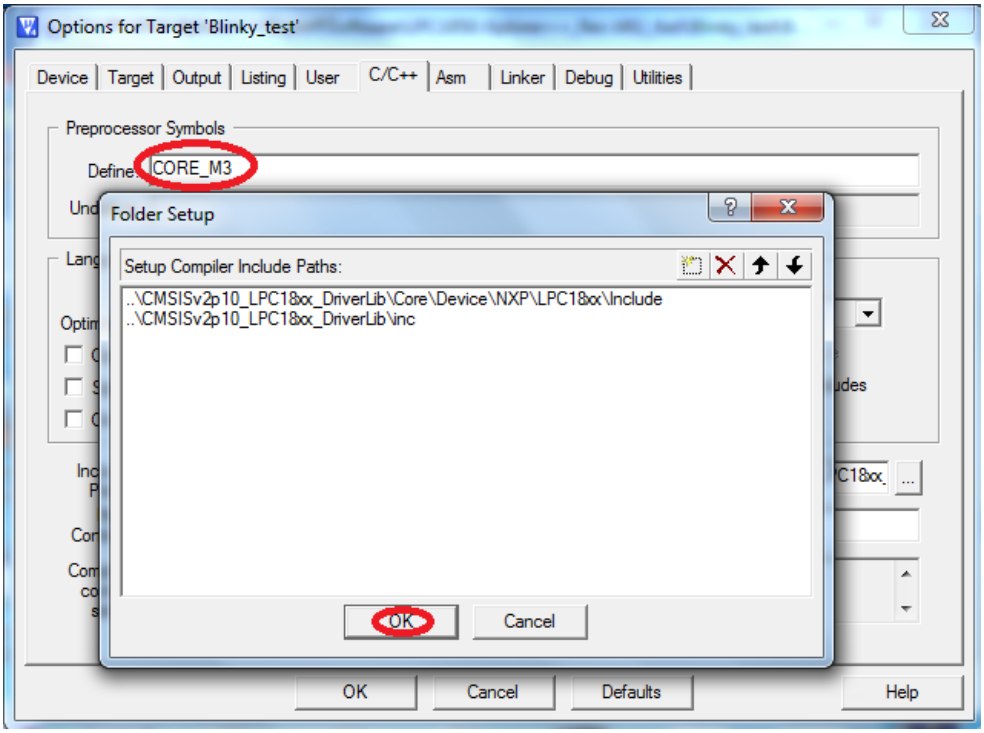


Fig.43

Step 20: In ‘Asm Option’, in Define type NO_CRP for assembly control symbols.

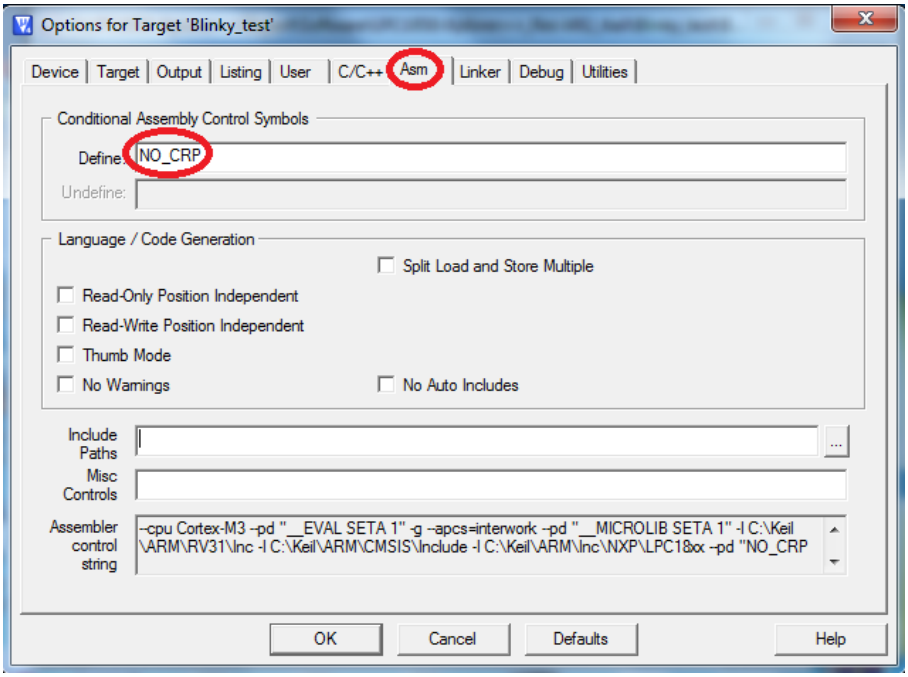


Fig.44

Step 21: In 'Debug Option', select 'ULINK2/ME Cortex Debugger' radio button and select 'Load Application at Startup' and 'Run to main()' check boxes, click on 'Initialization File': to select 'Internal SRAM.ini file', select 'Internal SRAM.ini file' and click Open as shown in the following image.

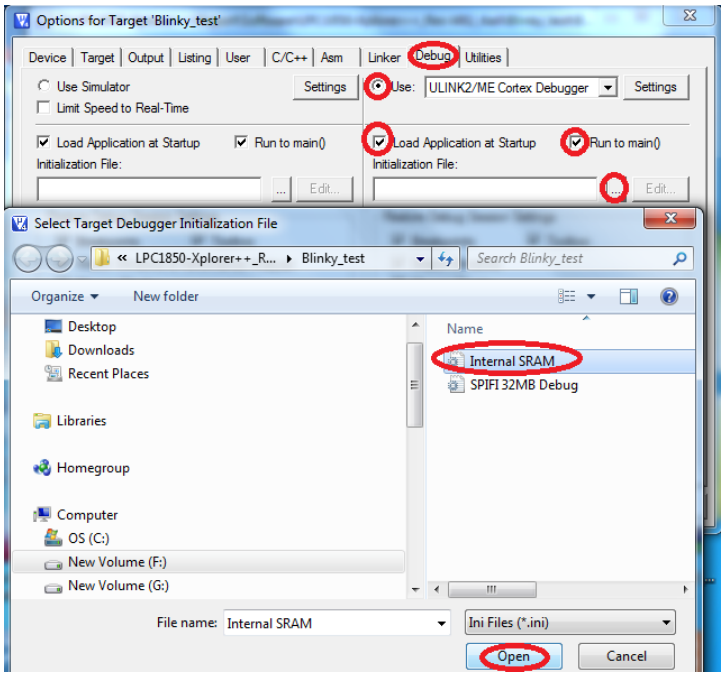


Fig.45

For 'External SPIFI Flash', make following changes as shown in the following image.

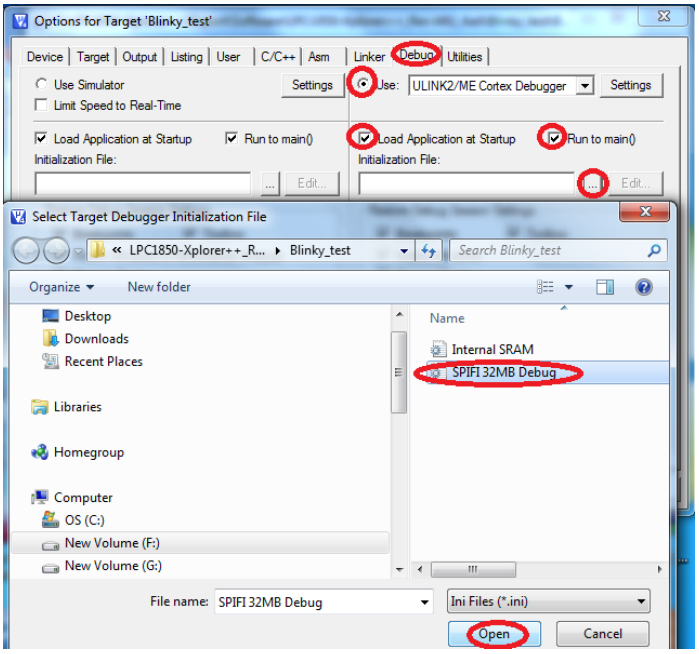


Fig.46

(Note: The 'ini file' will found at following path: ..\Blinky\Keil)

Step 22: In ‘Utilities Option’, select ‘Use External Tool for Flash Programming’.

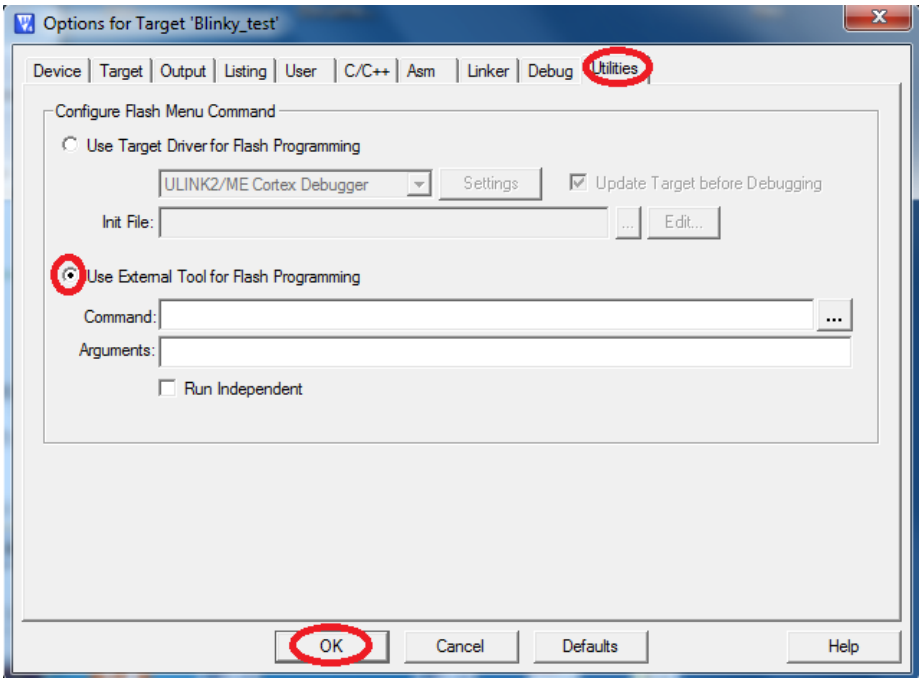


Fig.47

For ‘External SPIFI Flash’, select ‘Use Target Driver for Flash Programming’ as ‘ULINK2/ME Cortex Debugger’ and select ‘Update Target before Debugging’ check box then click ‘Setting Option’, remove the existing programing algorithm and click Add select ‘LPC18xx/43xx S25FL032 SPIFI Flash algorithm’ as shown in the following image.

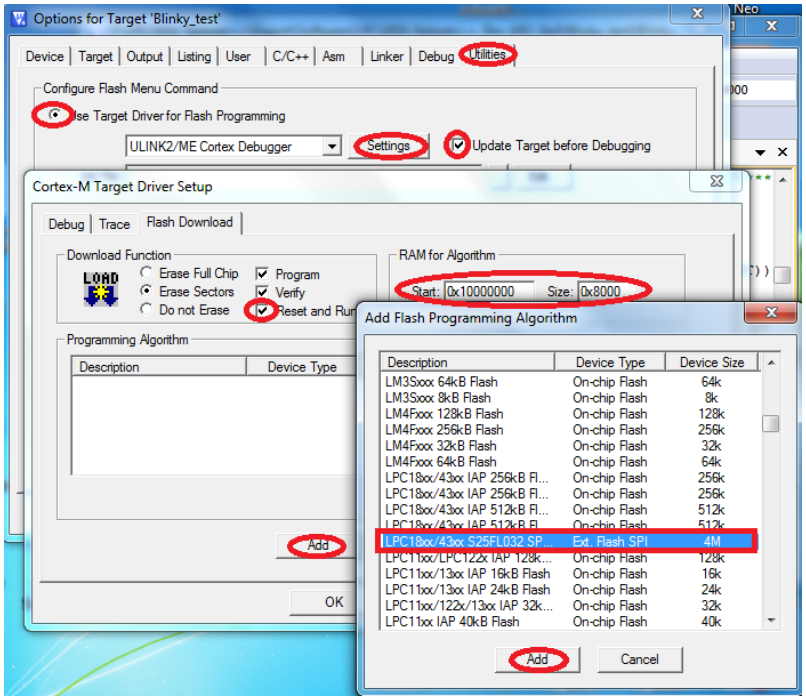


Fig.48

Step 23: In ‘Debug Option’, click on Edit... the ‘Internal SRAM.ini file’ will open in editor and OK.

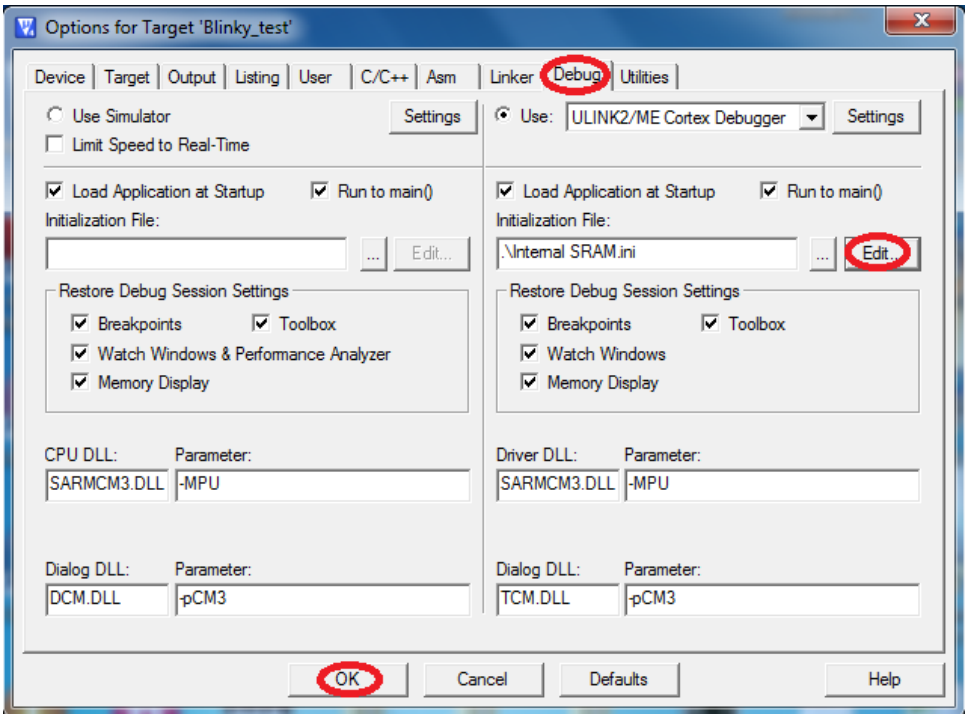


Fig.49

For ‘External SPIFI Flash’, click on Edit... the ‘SPIFI 32MB Debug.ini file’ will open in editor and click OK.

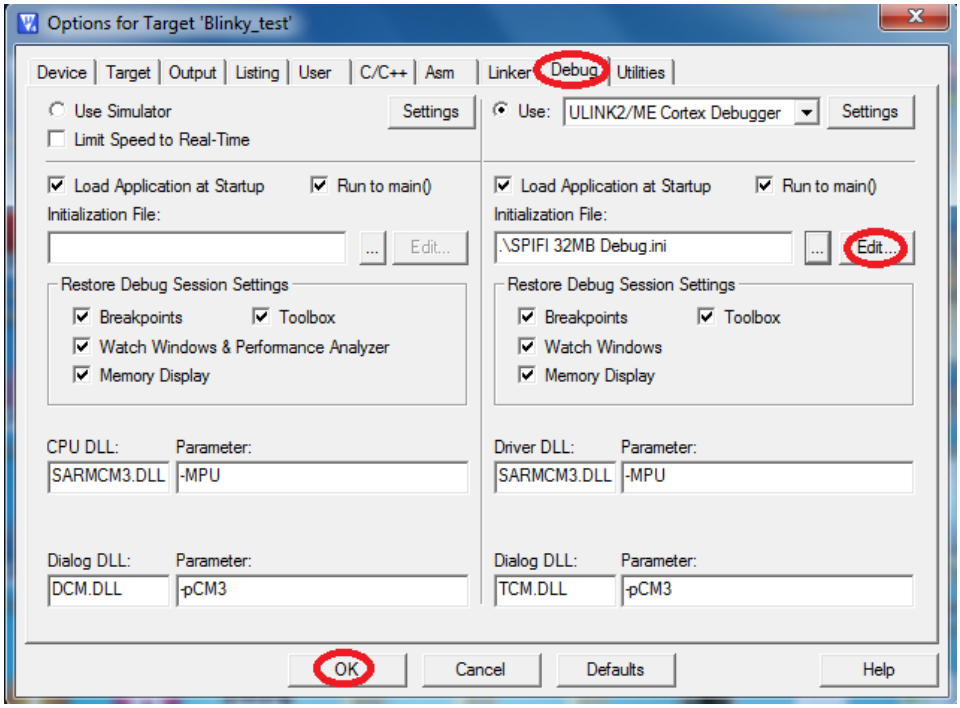


Fig.50

Step 24: Give the correct '.axf file' path in 'Internal SRAM.ini file' and file name should be same as shown in the following image.

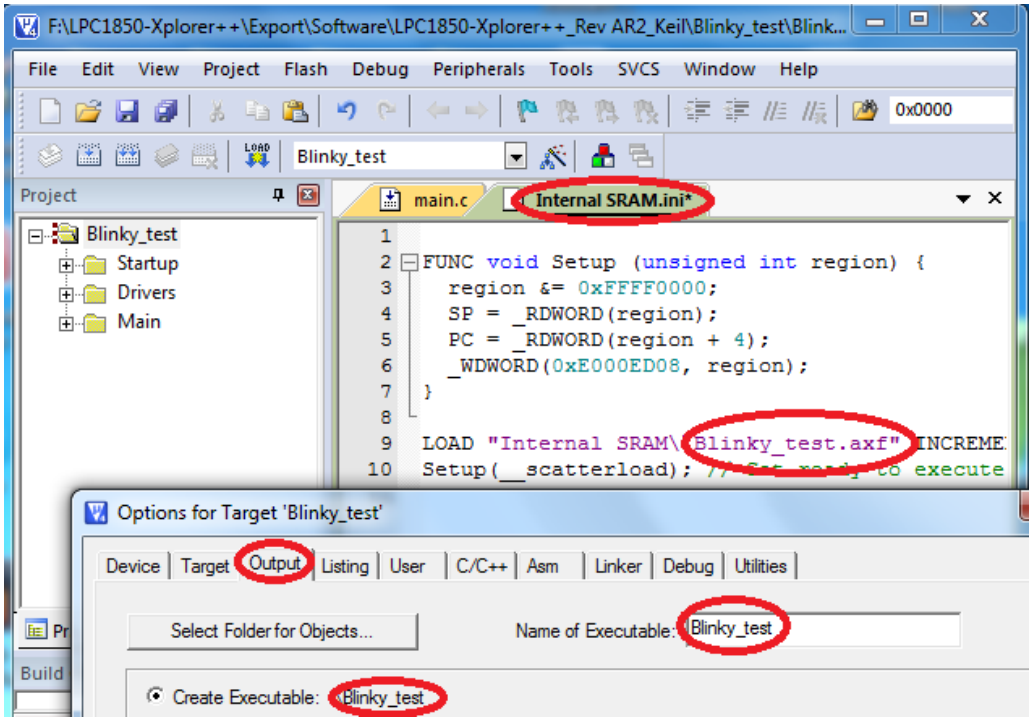


Fig.51

For 'External SPIFI Flash', Give the correct '.axf file' path in 'SPIFI 32MB Debug.ini file' and file name should be same as shown in the following image.

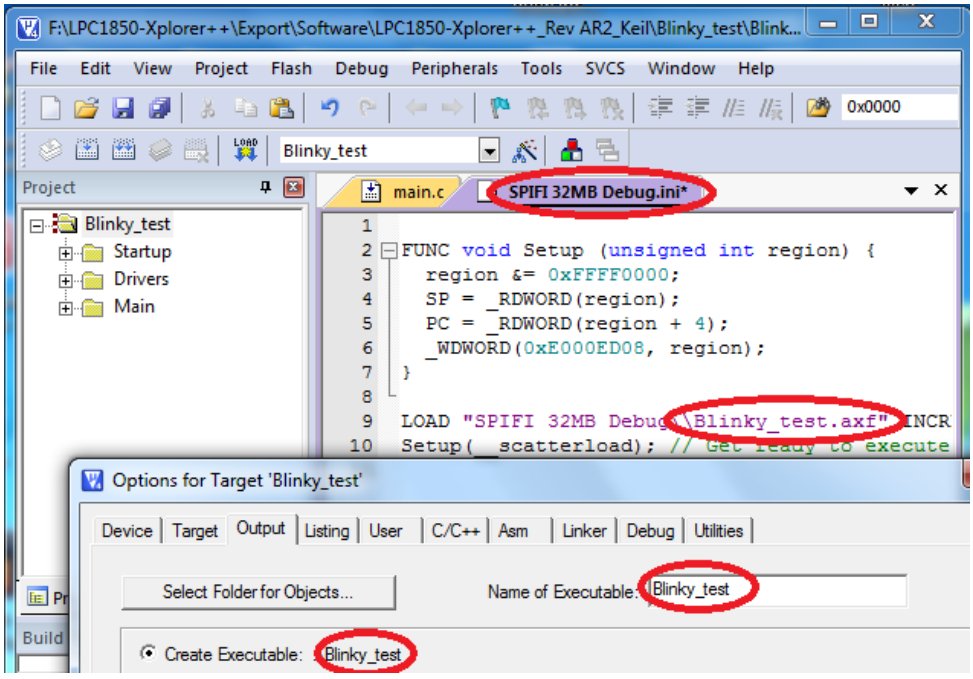


Fig.52

Step 25: Click on 'Build (F7)' to build a blinky_test project, the build should be error free.

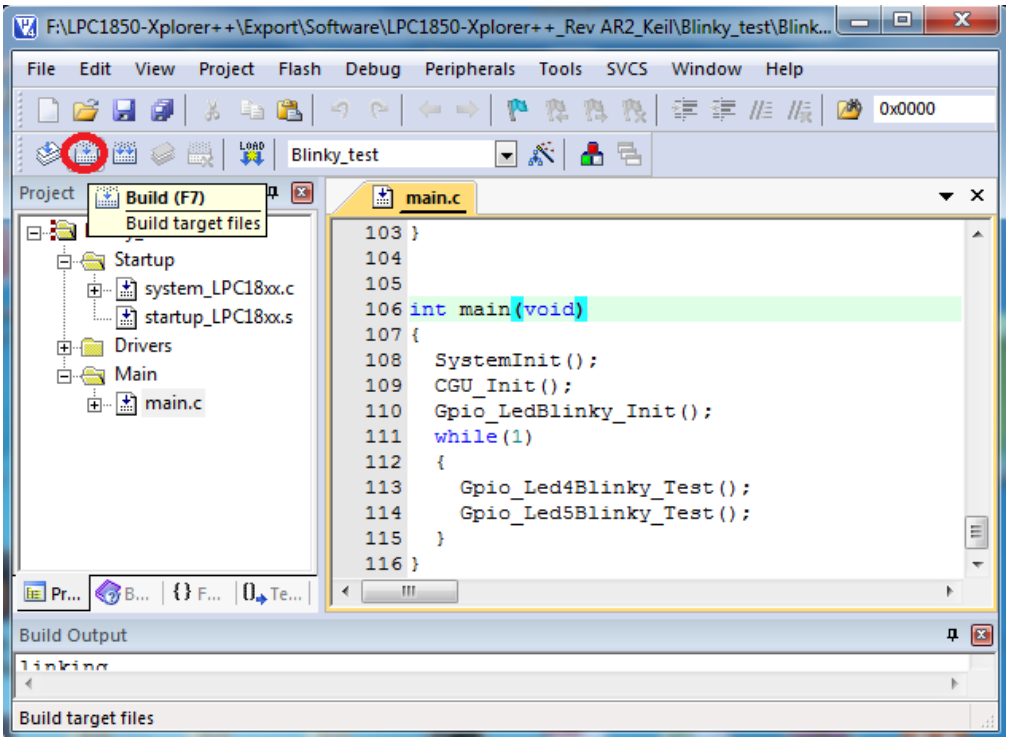


Fig.53

Step 26: Click on Debug -> 'Start/Stop Debug Session'.

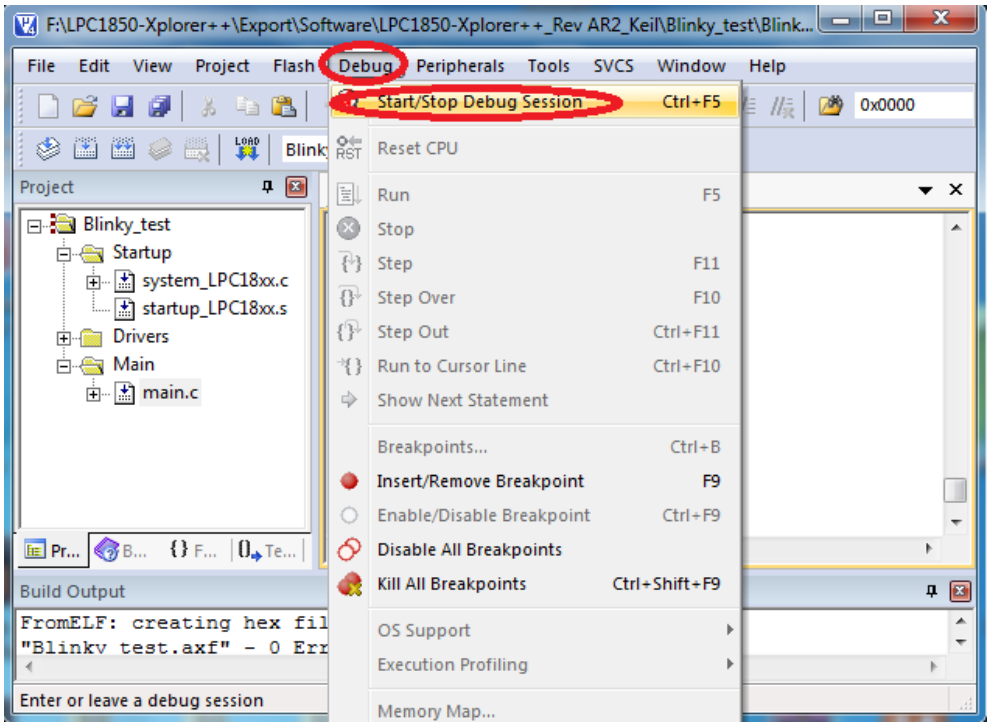


Fig.54

Step 27: Click on 'Run (F5)', on LPC185X-Xplorer++ board the LED4 and LED5 starts blinking.

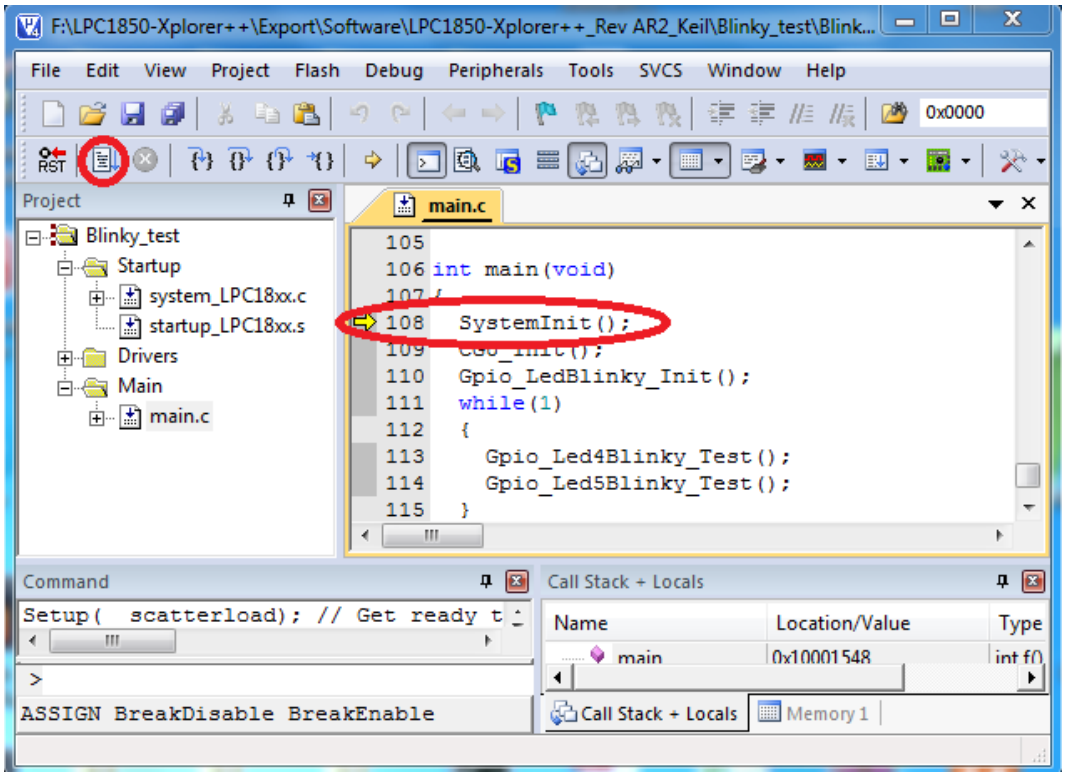


Fig.55

Step 28: Click on LOAD to download the Blinky_test.axf file into flash. Reset the board then the LED4 and LED5 starts blinking.

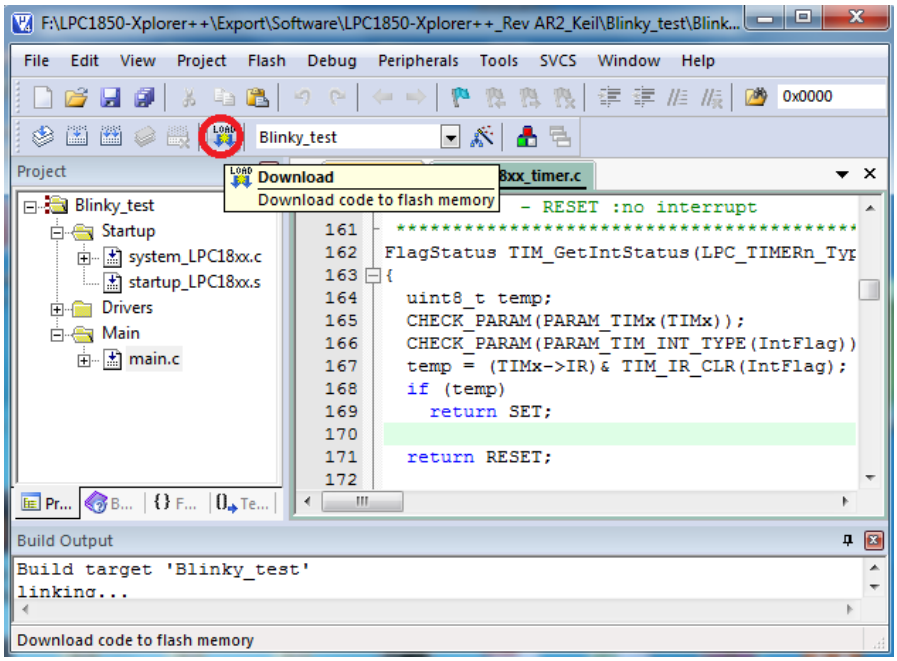


Fig.56

3.2 Executing the sample project in KEIL

Please note that the sample programs are available once the product is registered.

Steps to execute the sample project in 'Internal SRAM':

Step 1: Open project folder.
Step 2: Open project_name.uvproj file (Example: Blinky.uvproj.)

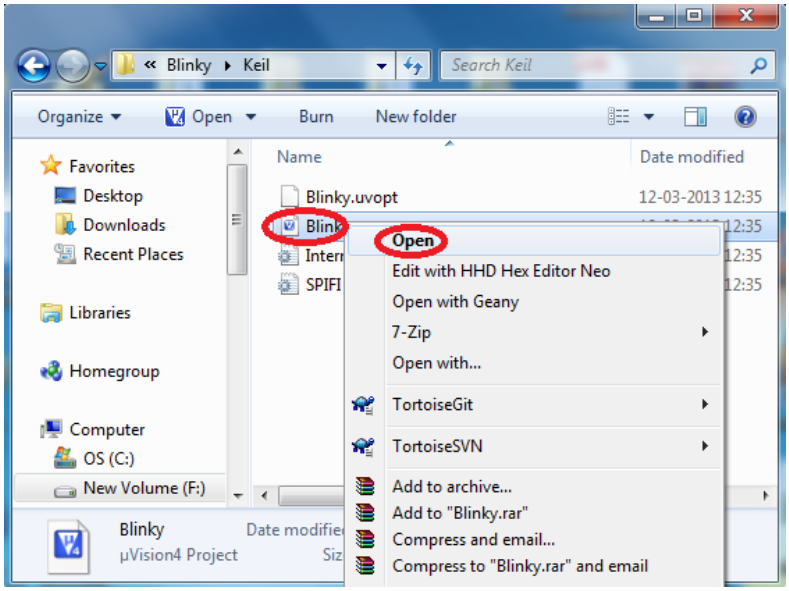


Fig.57

Step 3: This launches the IDE and double click on 'Main.c file', click on build, build must error free. Click on Debug -> 'Start/Stop Debug Session'.

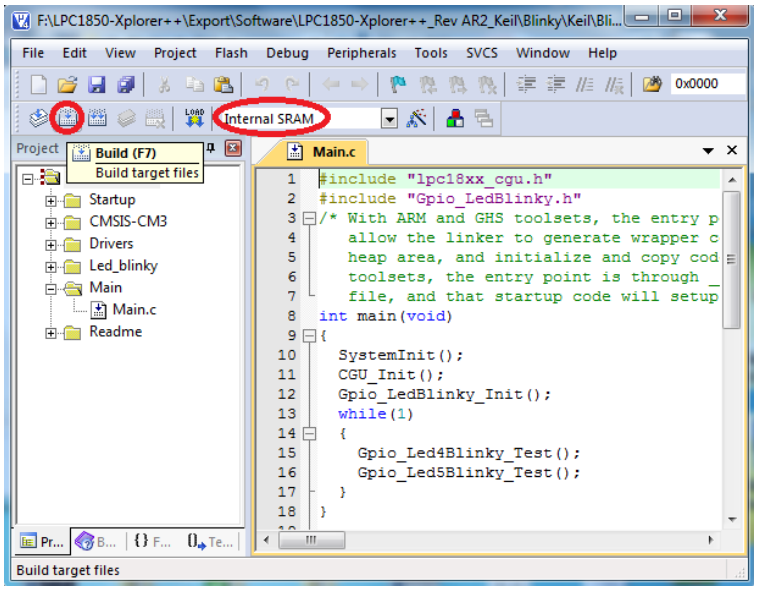


Fig.58

Step 4: Click Run (F5) to execute from the 'Internal SRAM', two LED's (LED4 and LED5) on LPC185X-Xplorer++ should blink.

Steps to execute the sample project in ‘External Quad Flash’ (SPIFI 32MB Debug):

Step 1: Select ‘SPIFI 32MB Debug Option’ and click on build as shown in the following image.

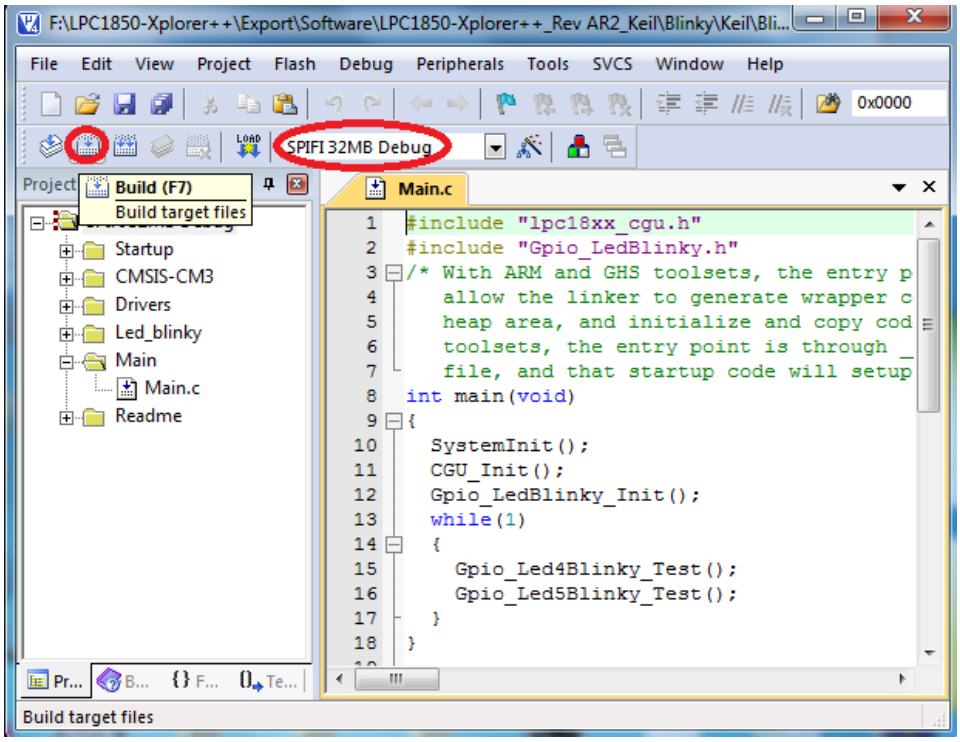


Fig.59

Step 2: The program can be debugged from the flash by clicking Debug -> ’Start/Stop Debug Session’, click Run (F5) to execute from the ‘External Quad Flash’ OR click on LOAD, the executable is loaded into ‘SPIFI 32MB flash’ then press RESET switch twice to run program from ‘External Quad Flash’, the two LED’s (LED4 and LED5) should start blinking on LPC185X-Xplorer++.

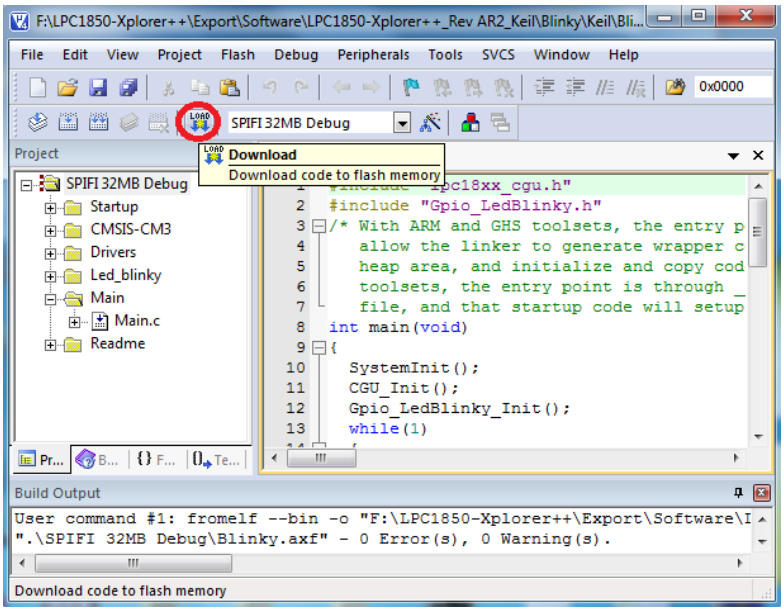


Fig.60

4.0 Creating bin file

Follow the below steps to create bin file.

Step1: Click on “Target Options”, Select “User” option, select Run #1 and type the format “fromelf --bin -o "$L@L.bin" "$L@L.axf" then click OK as shown in the following image.

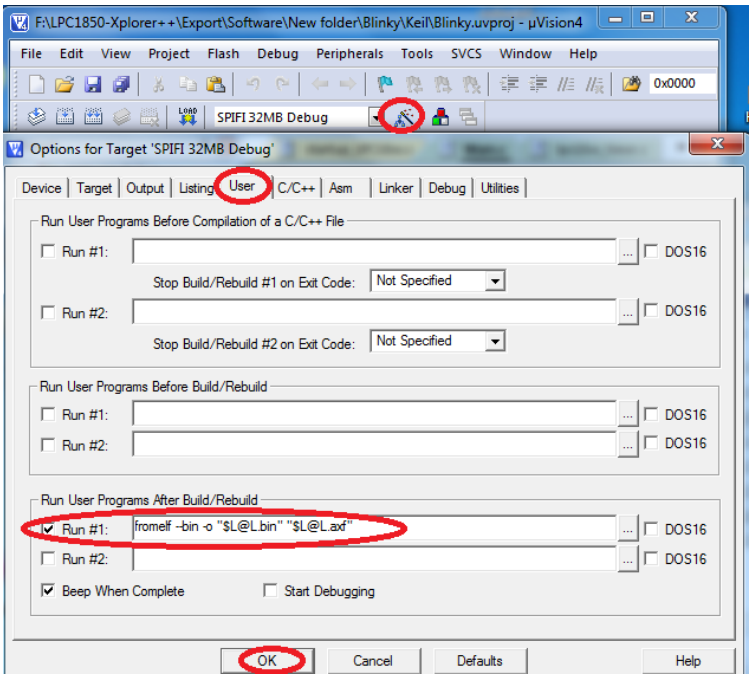


Fig.61

Step 2: Build the program as shown in the following image.

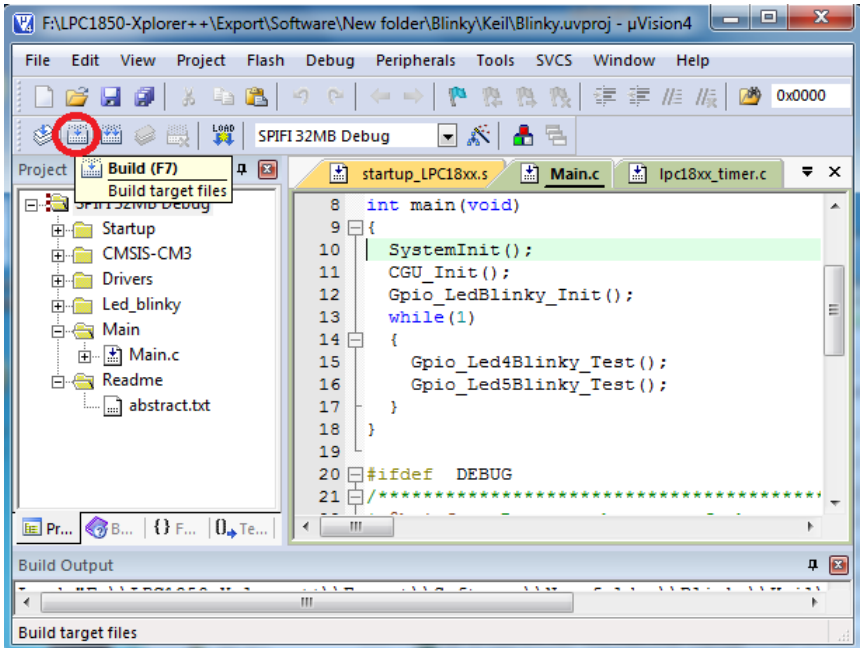


Fig.62

We can find the bin file in the FLASH folder of the project (Blinky\Keil\FLASH)

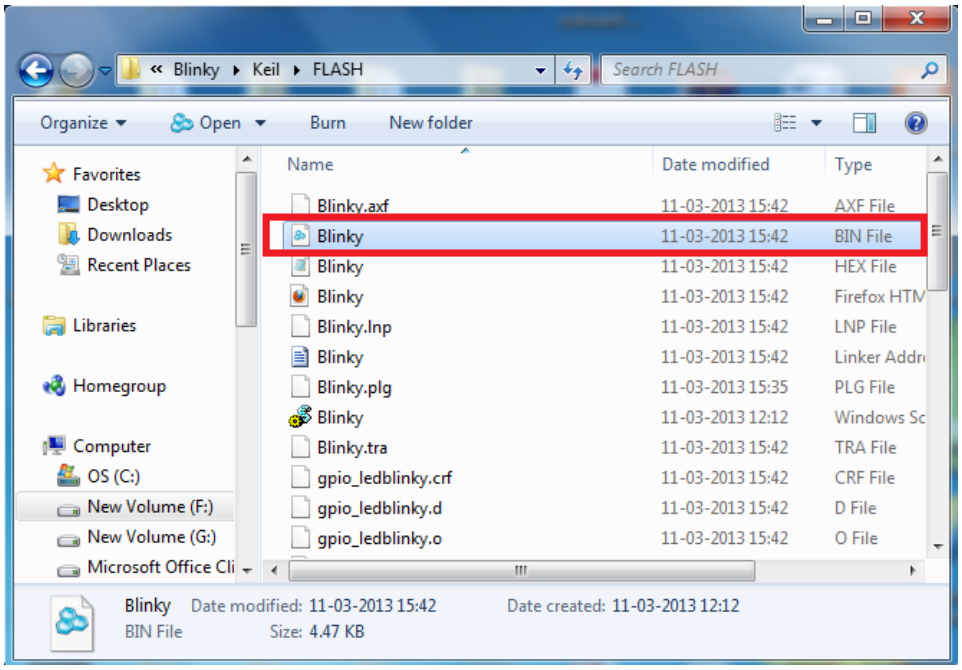


Fig.63

5.0 LPC185x-Xplorer++ Programming

5.1 Programming options

LPC185x-Xplorer++ can be programmed using the

- Coflash and ColinkEx

5.1.1 Programming LPC185x-Xplorer++ using Coflash and ColinkEx

Connect one end of ‘USB AM to Micro B’ cable to LPC185x-Xplorer++ board and other end to computer and connect one end of ‘USB AM to Micro B’ to ColinkEx and other end to computer as shown in the following image.

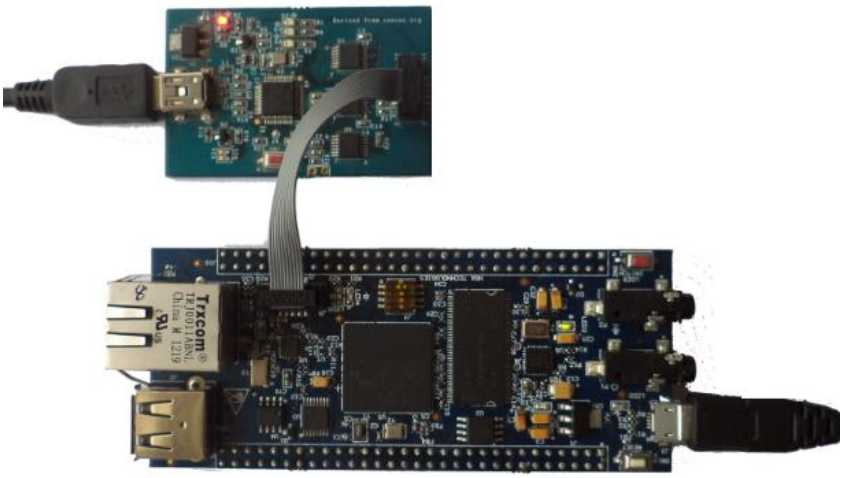


Fig.64

Step 1: Open Coflash, click on config, select controller and set all the fields as shown in the following image.

NOTE: To get the device LPC18xx-LPC43xx in the device list follows the steps given in this link

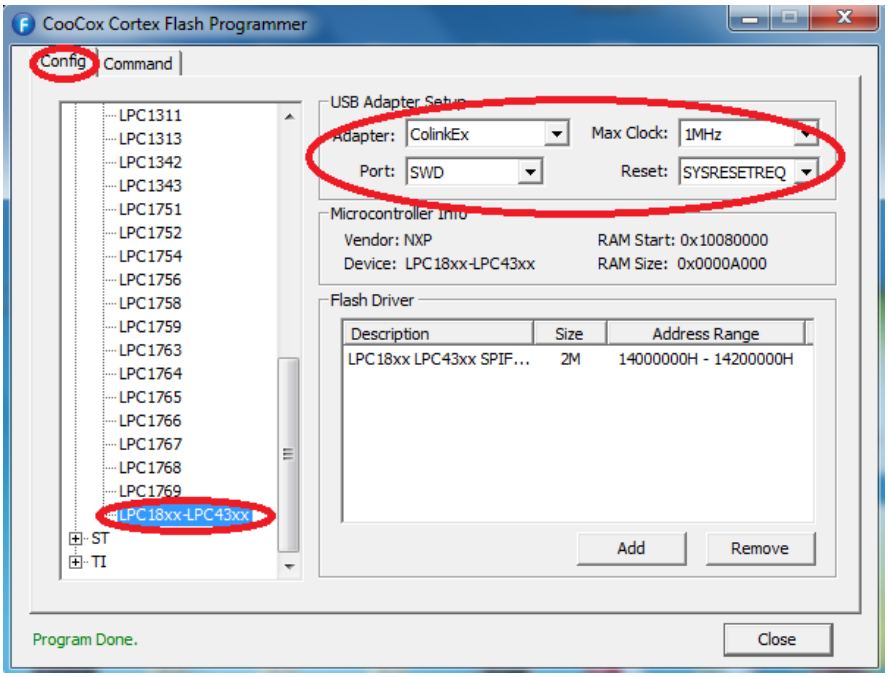


Fig.65

Step 2: Click on Command, select the bin file you want to download as shown in the following image

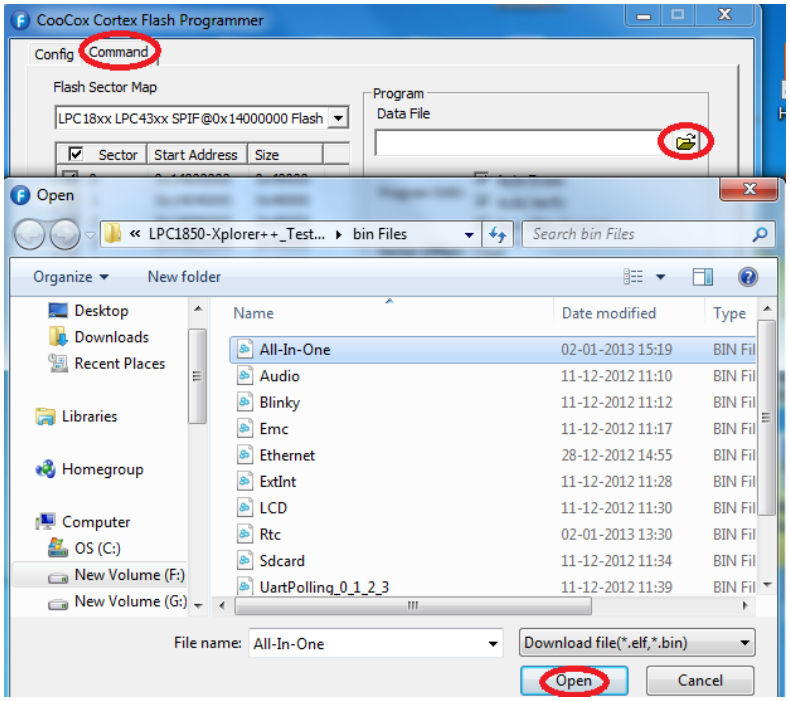


Fig.66

Step 3: Click on Program to load the bin file to the target board as shown in the following image

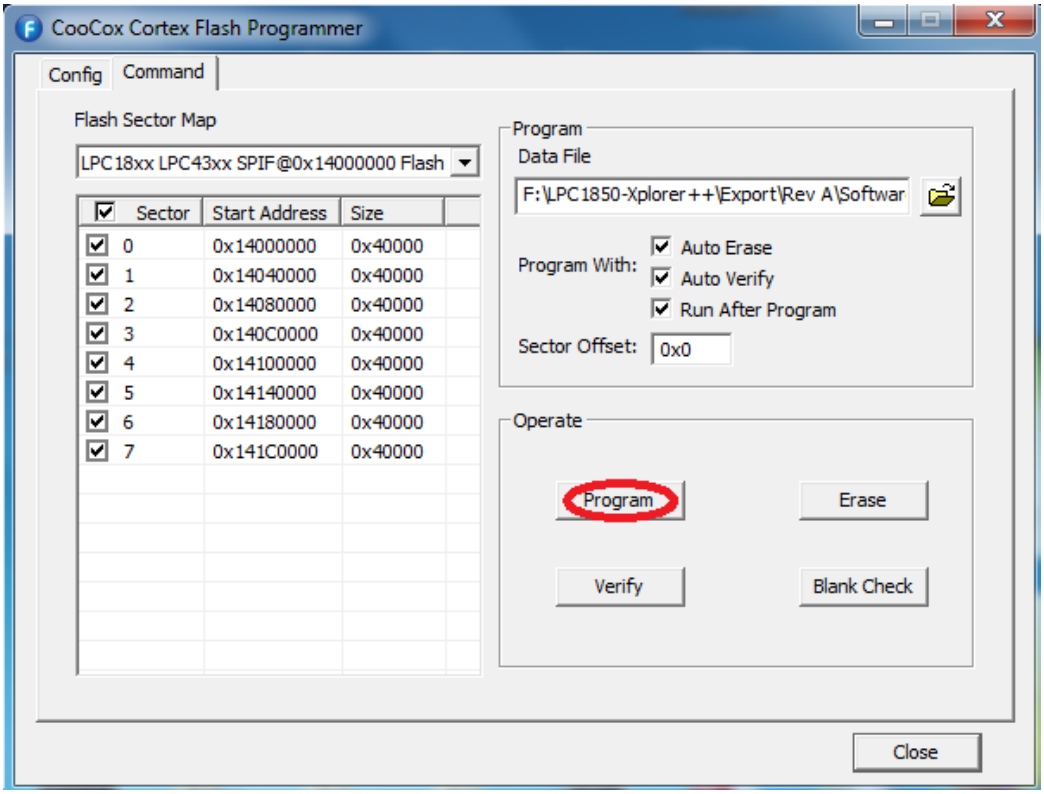


Fig.67

Once the programming is completed, reset the board to run the code.

6.0 Restoring LPC185X-Xplorer++ to Factory Defaults

6.1 ULINK2/ME and KEIL

To restore the factory defaults for the LPC185x-Xplorer++, User needs ULINK2/ME debugger to program the LPC185x-Xplorer++ to default firmware (i.e. **All-In-One.uvproj**).

Steps to restore the factory defaults for LPC185X-Xplorer++:

Step 1: Open **All-In-One** folder and double click on **All-In-One.uvproj** project.

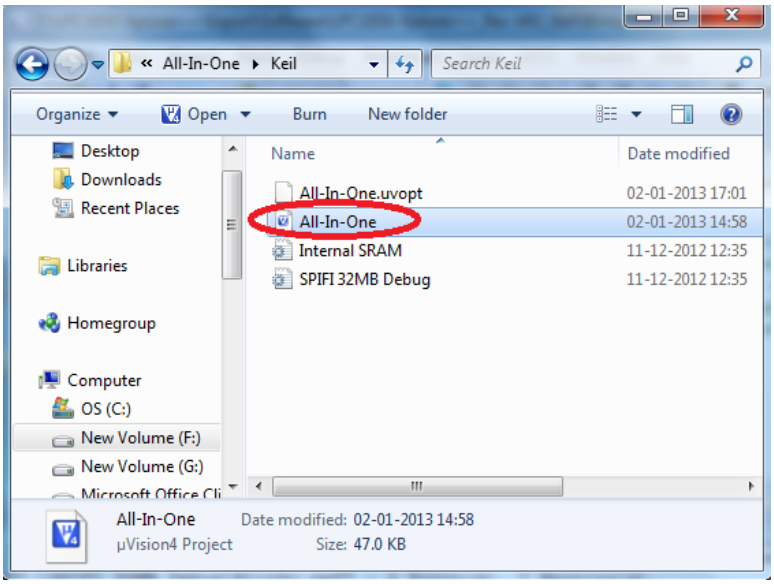


Fig.68

Step 2: Build and click on LOAD, the **All-In-One.axf** will flash on to SPIFI Flash. Now, RESET the board to restore the LPC185X-Xplorer++ to factory default.

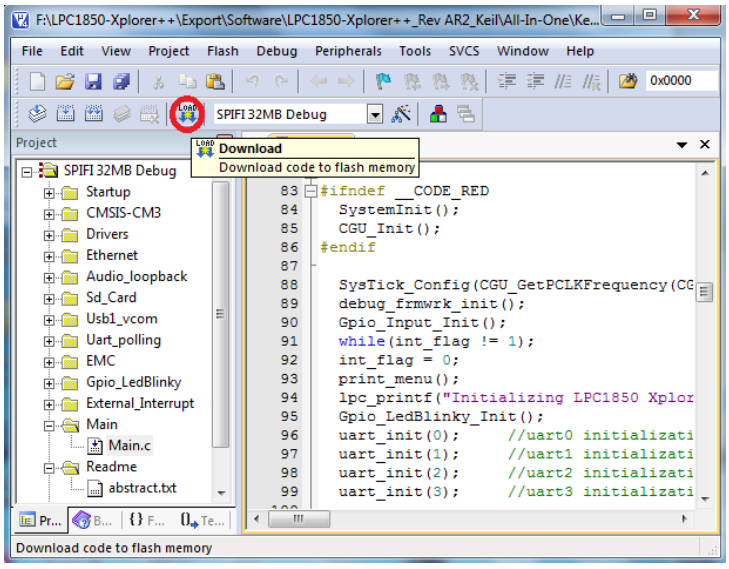


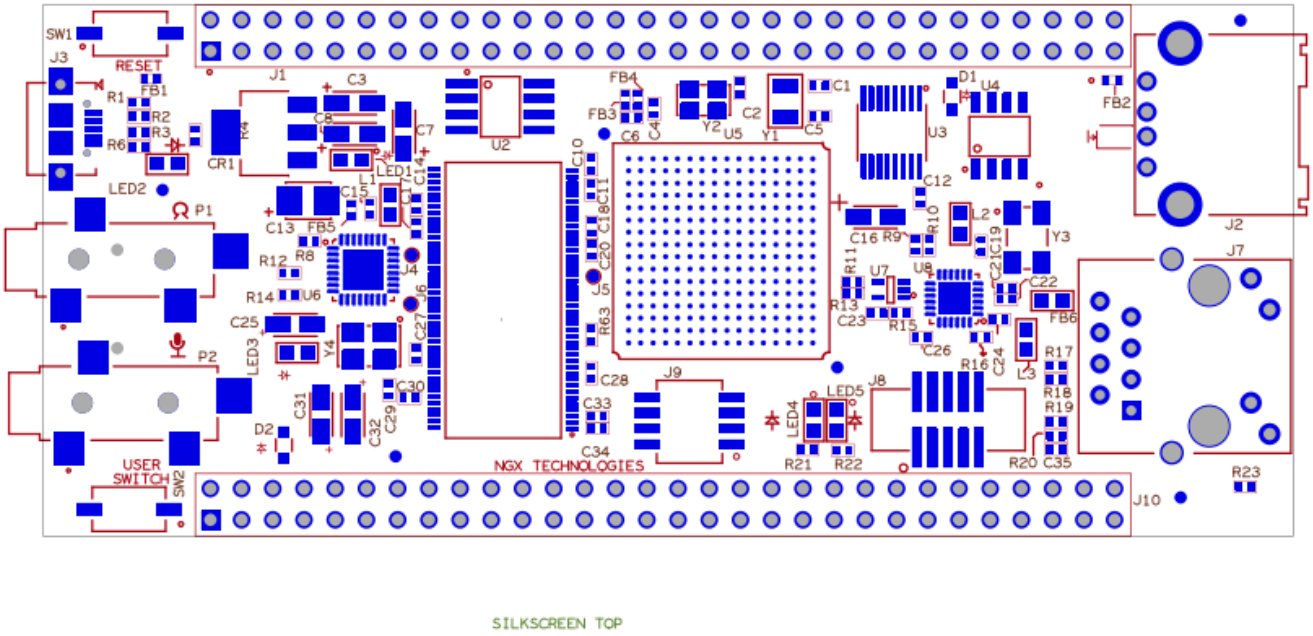
Fig.69

7.0 Schematic & Board Layout

7.1 Schematic

This manual will be periodically updated, please check our website for the latest documents. The Board schematic and sample code are available after the product has been registered on our website.

7.2 Board layout



SILKSCREEN TOP

Fig.70

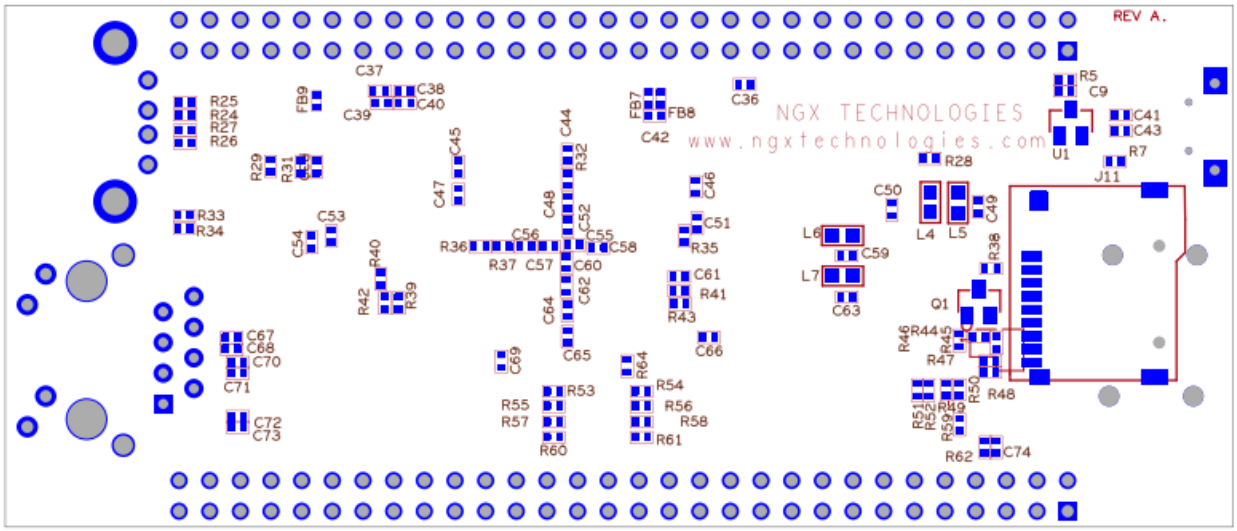


Fig. 71

8.0 CHANGE HISTORY

8.1 Change History

Rev	Changes	Date (dd/mm/yy)	By
1.0	Initial release of the manual	15/03/2013	Veeresh Tumbaragi

About this document:

Revision History

Version: V1.0 author: Veeresh Tumbaragi

Company Terms & Conditions

Legal

NGX Technologies Pvt. Ltd. provides the enclosed product(s) under the following conditions:

This evaluation board/kit is intended for use for ENGINEERING DEVELOPMENT, DEMONSTRATION, and EDUCATION OR EVALUATION PURPOSES ONLY and is not considered by NGX Technologies Pvt. Ltd to be a finished end-product fit for general consumer use. Persons handling the product(s) must have electronics training and observe good engineering practice standards. As such, the goods being provided are not intended to be complete in terms of required design-, marketing-, and/or manufacturing-related protective considerations, including product safety and environmental measures typically found in end products that incorporate such semiconductor components or circuit boards. This evaluation board/kit does not fall within the scope of the European Union directives regarding electromagnetic compatibility, restricted substances (RoHS), recycling (WEEE), FCC, CE or UL and therefore may not meet the technical requirements of these directives or other related directives.

The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies NGX Technologies from all claims arising from the handling or use of the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge.

EXCEPT TO THE EXTENT OF THE INDEMNITY SET FORTH ABOVE, NEITHER PARTY SHALL BE LIABLE TO THE OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

NGX Technologies currently deals with a variety of customers for products, and therefore our arrangement with the user is not exclusive. NGX Technologies assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein.

Please read the User's Guide and, specifically, the Warnings and Restrictions notice in the User's Guide prior to handling the product. This notice contains important safety information about temperatures and voltages.

No license is granted under any patent right or other intellectual property right of NGX Technologies covering or relating to any machine, process, or combination in which such NGX Technologies products or services might be or are used.

Disclaimers

Information in this document is believed to be reliable and accurate. However, NGX Technologies does not give any representations or warranties, expressed or implied, as to the completeness or accuracy of such information and shall have no liability for the consequences of use of such information.

NGX Technologies reserves the right to make changes to information published in this document, at any time and without notice, including without limitation specifications and product descriptions. This document replaces and supersedes all information supplied prior to the publication hereof.

Trademarks

All referenced trademarks, product names, brands and service names are the property of their respective owners.